

z/OS



Managed System Infrastructure for Operations Setting Up and Using

Version 1 Release 2

z/OS



Managed System Infrastructure for Operations Setting Up and Using

Version 1 Release 2

Note!

Before using this information and the product it supports, be sure to read the general information under "Notices" on page 255.

Second Edition (March 2002)

This edition applies to Version 1 Release 2 of z/OS (Program Number 5694-A01) and to all subsequent releases and modifications until otherwise indicated in new editions.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

A form for readers' comments appears at the back of this publication. If the form has been removed, address your comments to:

IBM Deutschland Entwicklung GmbH
Department 3248
Schoenaicher Strasse 220
D-71032 Boeblingen
Federal Republic of Germany

If you prefer to send comments electronically, use one of the following methods:

FAX (Germany): 07031 + 16-3456
FAX (Other Countries): (+49)+7031-16-3456
Internet: s390id@de.ibm.com

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright International Business Machines Corporation 2002. All rights reserved.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

About this book.	vii
Who should use this book	vii
How this book is organized	vii
Where to find more information	viii
Using LookAt to look up message explanations	viii
Accessing licensed books on the Web	ix

Part 1. Introducing Managed System Infrastructure for Operations 1

Chapter 1. General concept 3

Structure of msys for Operations and its position within the sysplex	4
Internal structure of msys for Operations	4
msys for Operations within the sysplex	4
Tasks and message automation	5
Understanding tasks	6
Automatic recovery and message automation	6

Chapter 2. Sysplex functions. 9

Managing couple data sets	9
Ensuring continuous availability of alternate couple data sets	9
INGPLEX CDS	10
Customization aspects	10
Managing the system logger	10
Resizing the LOGR couple data sets in case of directory shortage	11
Notifying the operator of incorrect share options	11
Customization aspects	11
Recovery functions	12
Resolving a system log failure	12
Resolving WTO(R) buffer shortages	12
Handling Long-Running Enqueues (ENQs)	13
Resolving Pending I/Os for Systems Being Removed From the Sysplex	13
Recovering Auxiliary Storage Shortage	13
Managing coupling facilities (INGCF command)	14
DRAIN	15
ENABLE	16
PATH	16
STRUCTURE	16
Customization	17
Miscellaneous	17
Recording IPL Information	17
Dump options	17

Part 2. Setting up msys for Operations 19

Chapter 3. Installing msys for Operations 21

Functional Prerequisites	21
--------------------------	----

Software Prerequisites	21
Hardware Prerequisites	22
Preparing the Hardware	23
Understanding the Hardware Interface	23
Preparing the Master Hardware Management Console	23
Preparing the Support Element	24
Planning for the domain and EMCS console names	25
Domain names	25
EMCS console names and the automation router task	25
Target libraries	26
Preparing the MVS system	27
Modifying the maximum number of language processor (REXX) environments for msys for Operations	27
Updating member SCHEDxx	28
Updating member MPFLSTxx	28
Message Automation	29
Updating the link list and APF authorizations with PROGxx	29
Updating member COMMNDxx	29
Preparing msys for Operations	29
Allocating data sets and VSAM clusters using job INGALLC0	29
Loading members of partitioned data sets	31
Preparing VTAM	32
Modifying the application (APPL) major node	32
Preparing the startup procedure and the initialization style sheet	33
Modifying the msys for Operations startup procedure	33
Customizing the initialization style sheet	34
Editing the customization member AOFCUST	35
AUTO section – switching functions On and Off	36
COMMON section – common definitions	37
PAGE section – predefined and dynamically allocated local page data sets	38
ENQ section – resources to be monitored and jobs to be cancelled	40
CDS section – spare volumes for CDS recovery	43
HW section – hardware configuration	44
IXC102A section – hardware commands	46
WTOBUF section – jobs to be cancelled in case of buffer shortage	47

Chapter 4. Making security definitions 49

Activating security classes	49
Defining operators, passwords, and logon attributes	49
Defining operator identifiers and passwords	49
Defining operator logon attributes in the NVSS segment of an SAF product	50
Command authorization	51
Overview	51
Command authorization using an SAF product	52

Additional recommendations for command authorization	55
Determining the user identity used for authority checking commands	56
Understanding security for specific commands	58
Controlling access to data sets and members	63
Data set security.	63
NVSS READSEC and WRITESEC commands	66
Controlling Access to the Processor Hardware Functions	67
Enabling NVSS	67
Access to the CPCs	67
Levels of CPC Access	68
Defining the CPC Access Level	68

Chapter 5. Activating msys for Operations 69

Starting msys for Operations Using the MSOAPROC Startup Procedure	69
Logging On to msys for Operations	70

Chapter 6. Configuring msys for Operations for Your Environment . . . 71

Defining Passwords for VSAM Databases	71
Printing the Network Log and Trace Log	71

Part 3. Using the operator interface of msys for Operations 73

Chapter 7. Logging on to msys for Operations 75

Chapter 8. The message display screen 77

Session identification line.	77
Message area	78
Reply messages	78
Held messages	79
Windowed responses	79
Response area	79
Command entry area	80

Chapter 9. Issuing commands. 81

Repeating commands	81
Issuing MVS system commands	81

Chapter 10. Moving between the components and using function keys . 83

Moving between the Components	83
Using program function keys	83
Listing PF and PA keys	83

Chapter 11. Using the netlog 85

Displaying the netlog	85
Structure of the log entries	85
Log browse filtering	87

Chapter 12. Getting online help 89

Examples	89
--------------------	----

Part 4. Command reference 91

Chapter 13. General commands 93

ALL (BROWSE, WINDOW)	93
AUTOWRAP	93
BACK (BROWSE, HELP, WINDOW)	93
BLOG	94
BOTTOM (BROWSE, HELP).	94
BROWSE	94
DEFAULTS	95
DISPFK.	95
END (BROWSE, HELP)	95
FIND (BROWSE)	96
FIND (WINDOW)	96
FORWARD	97
HELP	97
INDEX	97
INPUT	98
LIST	98
LISTA	98
LOCATE (BROWSE)	99
LOGOFF	99
MSG.	99
MVS	100
OVERRIDE	100
REPEAT (BROWSE)	100
RETRIEVE	101
RETURN (BROWSE, HELP)	101
ROLL	101
SET.	101
TOP (BROWSE, HELP)	102
WHO	102
WINDOW	102

Chapter 14. Sysplex-related commands 105

INGCF	106
DRAIN	111
ENABLE	116
PATH	120
STRUCTURE	121
INGCFL	123
Purpose	123
Syntax.	123
Parameters	124
INGPLEX	124
CDS	128
SYStem	133
CONsole	135
IPL	137
SDUMP	138
SVCdump	141
SLIP	144

Chapter 15. Debugging and supporting commands 147

ACF	147
---------------	-----

AOCTRACE	147
DISPACF	149
DISPAOPS	149
DISPASF	150
DISPATHR	151
DISPERRS	152
DISPFLGS	153
DISPMGS	153
DISPWTOR	154
INGAUTO	155
INGCUST	155

Part 5. Setup reference 157

Chapter 16. msys for Operations

Definition Statements Reference . . . 159

Statement Formats.	159
Syntax Conventions for Definition Statements	159
AUTH.	160
OPERATOR	161
OPTIONS	162
PROFILE	169
PROFLEN	170
TRANSTBL	171

Part 6. Automation-related messages 173

Chapter 17. Messages. 175

Part 7. Appendices 189

Appendix A. Making security definitions using the command authorization table 191

Defining security using msys for Operations definitions	191
Overview of operator security.	191
Using msys for Operations for password authorization	192
Using msys for Operations without password authorization	192
Operator logon attributes	192
Defining operator attributes in msys for Operations profiles	193
Dynamically adding or deleting operators.	194

Command authorization.	194
Controlling access to data sets and members	213

Appendix B. Return codes. 217

Macro return codes	217
Macro DSICES return codes in Register 15.	217
Macro DSIDKS return codes in Register 15	218
Macro DSIMQS return codes in Register 15	220
Macro DSIPRS return codes in Register 15.	220
Macro DSIPSS return codes in Register 15	221
Macro DSIPUSH return codes in Register 15	222
Macro DSIZCSMS return codes in Register 15	223
Macro DSIZCSMS minor return codes in Register 0	223
Macro DSIZVSMS return codes in Register 15	223
Macro DSIZVSMS minor return codes in Register 0	224
VIEW and BROWSE return codes	224

Appendix C. Coexistence of msys for Operations and SA OS/390 V1R3. . . 227

Appendix D. Response Messages, Error Strings, Condition Codes. . . . 229

Response Messages (AOFA0000 – AOFA0018)	229
Error Strings	232
Condition Codes	233
Hardware Communication Task Condition Codes "00B00xxx"	233
SNMP Data Exchange Services "0B100xxx"	235
SNMP Command Services "0B200xxx"	236
BCP Internal Interface Transport Services "0Bx00xxx"	237

Appendix E. Sense Codes, Hardware Object Status Summary 241

Sense Codes.	241
Hardware Object Status Summary	249

Glossary 251

Notices 255

Programming interface information	256
Trademarks	256

Index 257

About this book

This book describes Managed System Infrastructure for Operations (msys for Operations). msys for Operations provides a simplified operating environment for Parallel Sysplexes. The simplification mainly relates to two areas, automation of recovery actions and a command interface that allows to perform complex tasks by simply pressing some function keys in a certain order.

Note: The sections marked with revision bar ‘|’ describe enhanced automation functions shipped with SPE OW50146 (03/2002). This functionality is only available if you have installed OW50146.

Who should use this book

This book is intended for sysplex operators and system programmers. The sysplex operators will use the enhanced command interface of msys for Operations for their daily work; the system programmers are primarily responsible for installation and customization of the product, but will also make use of the command interface.

Both types of users are expected to have a basic knowledge of a Parallel Sysplex.

How this book is organized

This book consists of five parts:

- Part 1, “Introducing Managed System Infrastructure for Operations” on page 1 describes the general concept and the functional scope of msys for Operations, including brief explanations of the relevant sysplex terms and concepts. It is intended as a general introduction for both operators and system programmers. System programmers can also use it to decide which automatic recovery actions are to be enabled.
- Part 2, “Setting up msys for Operations” on page 19 describes the installation and customization of msys for Operations. It also describes how to protect system resources from unauthorized use. This part is of interest mainly to the system programmer.
- Part 3, “Using the operator interface of msys for Operations” on page 73 describes the operator interface of msys for Operations.
- Part 4, “Command reference” on page 91 contains descriptions of all commands that are available with msys for Operations. This part is divided into three chapters:
 - Chapter 13, “General commands” on page 93 contains the commands of the operator interface.
 - Chapter 14, “Sysplex-related commands” on page 105 describes the commands for sysplex management.
 - Chapter 15, “Debugging and supporting commands” on page 147 describes commands that are used for problem determination. These commands will mainly be used by system programmers.
- Part 5, “Setup reference” on page 157 contains information for the system programmer about additional security options and the configuration statements of NVSS.

Where to find more information

Title	Order Number
<i>z/OS MVS Setting Up a Sysplex</i>	SA22-7625
<i>z/OS MVS Programming: Sysplex Services Guide</i>	SA22-7617

z/OS msys for Operations Homepage

For the latest news on msys for Operations, visit the msys for Operations homepage at <http://www.ibm.com/servers/eserver/zseries/msys/msysops>

Using LookAt to look up message explanations

LookAt is an online facility that allows you to look up explanations for most of the z/OS, z/VM, and VSE messages you encounter, as well as system abends and some codes. Using LookAt to find information is faster than a conventional search because in most cases LookAt goes directly to the message explanation.

You can access LookAt from the Internet at:

<http://www.ibm.com/servers/eserver/zseries/zos/bkserv/lookat/>

or from anywhere in z/OS where you can access a TSO command line (for example, TSO prompt, ISPF, z/OS UNIX System Services running OMVS). You can also download code from the *z/OS Collection* (SK3T-4269) and the LookAt Web site so you can access LookAt from a PalmPilot (Palm VIIx suggested).

To use LookAt on the Internet to find a message explanation, go to the LookAt Web site and simply enter the message identifier (for example, \$HASP701 or \$HASP*). You can select a specific release to narrow your search.

To use LookAt as a TSO command, you must have LookAt installed on your host system. You can obtain the LookAt code for TSO from a disk on your *z/OS Collection* (SK3T-4269) or from the LookAt Web site. To obtain the code from the LookAt Web site, do the following:

1. Go to <http://www.ibm.com/servers/eserver/zseries/zos/bkserv/lookat/>.
2. Click **News**.
3. Scroll to **Download LookAt Code for TSO and z/VM**.
4. Click the ftp link, which will take you to a list of operating systems. Click the appropriate operating system. Then click the appropriate release.
5. Open the **lookat.me** file and follow its detailed instructions.

After you have LookAt installed, you can access a message explanation from a TSO command line by entering: **lookat** *message-id*. LookAt will display the message explanation for the message requested.

Note: Some messages have information in more than one book. For example, IEC192I can be found in *z/OS MVS System Messages, Vol 7 (IEB-IEE)* and also in *z/OS MVS Routing and Descriptor Codes*. For such messages, LookAt displays a list of books in which the message appears. You can then select one of the books to view the message explanation.

Accessing licensed books on the Web

OS/390 licensed documentation in PDF format is available on the Internet at the IBM Resource Link Web site at:

<http://www.ibm.com/servers/resourceLink>

Licensed books are available only to customers with a OS/390 license. Access to these books requires an IBM Resource Link Web userid and password, and a key code. With your OS/390 order you received a memo that includes this key code.

To obtain your IBM Resource Link Web userid and password log on to:

<http://www.ibm.com/servers/resourceLink>

To register for access to the z/OS licensed books:

1. Log on to Resource Link using your Resource Link userid and password.
2. Click on **User Profiles** located on the left-hand navigation bar.
3. Click on **Access Profile**.
4. Click on **Request Access to Licensed books**.
5. Supply your key code where requested and click on the **Submit** button.

If you supplied the correct key code you will receive confirmation that your request is being processed. After your request is processed you will receive an e-mail confirmation.

Note: You cannot access the z/OS licensed books unless you have registered for access to them and received an e-mail confirmation informing you that your request has been processed.

To access the licensed books:

1. Log on to Resource Link using your Resource Link userid and password.
2. Click on **Library**.
3. Click on **zSeries**.
4. Click on **Software**.
5. Click on **OS/390**.
6. Access the licensed book by selecting the appropriate element.

Part 1. Introducing Managed System Infrastructure for Operations

This part describes the goals and scope of the functions that are offered by Managed System Infrastructure for Operations (msys for Operations). It is intended for both the system programmer and the operator.

Note: The sections marked with revision bar ‘|’ describe enhanced automation functions shipped with SPE OW50146 (03/2002). This functionality is only available if you have installed OW50146.

Chapter 1. General concept

msys for Operations provides a simplified environment for operating Parallel Sysplexes. Managing a Parallel Sysplex, while easier than administrating the member systems independently, is still a rather complex task. Recovery and maintenance actions may require the operator to enter long chains of complicated commands. Furthermore, these chains may branch depending on the status of the resources to be managed. Even gathering the necessary information can be difficult.

The following examples can illustrate this point:

1. After a switch from a primary to the alternate couple data set (CDS), the system issues the message IXC263I, which indicates that an alternate CDS is no longer available. Since this would entail a single point of failure, the operator must provide a new alternate CDS. In order to allocate and format the new alternate CDS, the operator must call the formatting tool for CDSs. In this call several parameters must be specified, and it is essential that the parameter values for the new alternate CDS are not smaller than those for the old alternate CDS (which is now the primary). After that, the new alternate CDS must be introduced to the system (with the SETXCF COUPLE,ACOUPLE command).
2. In order to remove a coupling facility (CF) from the sysplex the sender paths of all systems to that CF must be set OFFLINE. Currently the operator must do this separately for every system. Furthermore, there is the risk that the operator may erroneously disable the sender paths of the backup CF which serves to maintain operations during the removal of the other CF, in which case the sysplex will go down.

msys for Operations reduces these complexities of sysplex operating as follows:

- A number of recovery actions can be completely automated. These include:
 - Creating or recreating missing alternate couple data sets
msys for Operations reacts to message IXC263I, calls the formatting tool with the parameters of the current primary CDS, and introduces the new alternate CDS to the system. No operator intervention is required.
 - Expanding the couple data sets for the system logger in case of a directory shortage
 - Resolving WTO(R) buffer shortages
 - System log recovery

Each automatic recovery action can be enabled or disabled separately. A number of them can be customized. By default, all actions are disabled. Enabling, disabling, and customizing is performed by editing a special data set member. This member and its entries are described in detail in “Editing the customization member AOF CUST” on page 35.

- msys for Operations provides two global commands (INGPLEX and INGCF) that:
 - Display comprehensive information about sysplex components (for example, couple data sets, policies, coupling facilities, structures, members of the sysplex) in a structured way

- Allow the operator to perform complex command sequences on the basis of the displayed information by simply pressing some function keys (for example, drain a coupling facility, or reintegrate a coupling facility into the sysplex)

One step of the draining process is to set the sender paths of the respective CF offline, as described in example 2 on page 3. This can be done with msys for Operations for all members of the sysplex with one key stroke. Also, there is no risk of confusing the CF to be removed with the backup CF.

The INGPLEX and INGCF commands support actions that have an impact on the sysplex configuration. You can control access to these actions through a security product such as RACF. For details, see “Command authorization” on page 51.

For a short explanation of the relevant sysplex concepts and a more detailed description of the functional scope of msys for Operations, see Chapter 2, “Sysplex functions” on page 9.

Structure of msys for Operations and its position within the sysplex

This section briefly sketches the internal structure of msys for Operations and its position within the sysplex.

Internal structure of msys for Operations

msys for Operations combines functions of two licensed products, NetView for OS/390 and System Automation for OS/390 (SA OS/390). The SA OS/390 part of msys for Operations runs on top of the NetView part. The NetView part supplies the infrastructure with basic services such as the command interface, triggering of automated recovery, and logging. It will be referred to as NVSS (NetView System Services) in this book. For more information on NVSS, see “Tasks and message automation” on page 5. The SA OS/390 part provides the sysplex specific recovery functions and commands. It will be referred to as *Sysplex Functions*.

msys for Operations contains a subset of the functions offered by the licensed products. You can migrate to the licensed products from msys for Operations. The relation between the licensed products is similar to that of the two parts of msys for Operations. A licensed NetView is a prerequisite for the automation functions of the licensed SA OS/390. Note that the origin of msys for Operations from two licensed products is still visible in its panels and help texts. Therefore, you may encounter ‘NetView’ or ‘SA OS/390’ where you would expect ‘msys for Operations’.

msys for Operations within the sysplex

The following figure shows a sample sysplex with msys for Operations installed:

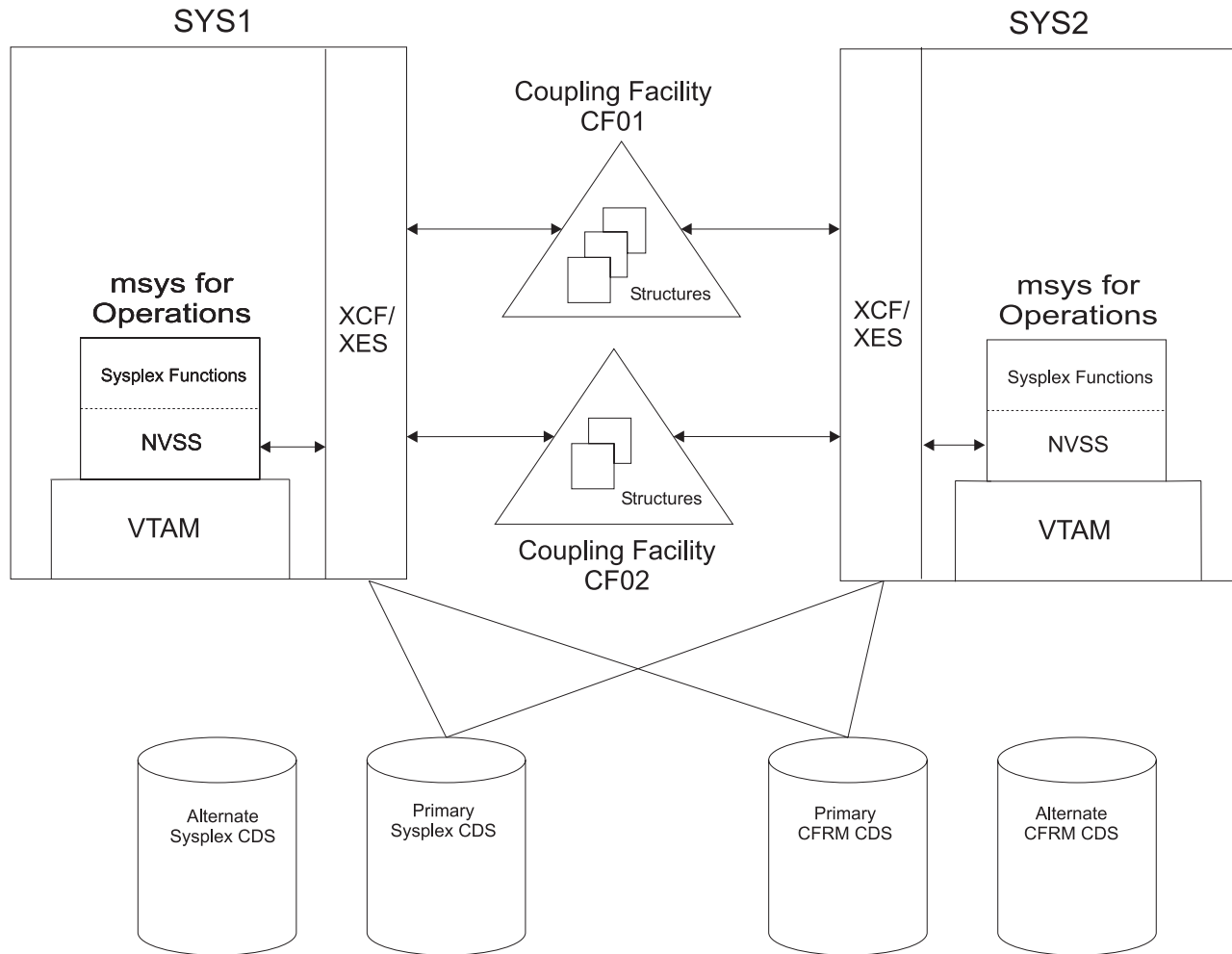


Figure 1. *msys for Operations* in a simple Parallel Sysplex configuration

This Parallel Sysplex consists of two MVS images and two coupling facilities. XCF (Cross-system Coupling Facility) and XES (Cross-system Extended Services) are the sysplex components of MVS. XCF supplies the basic sysplex services, XES manages the coupling facilities and structures. The sysplex components of both MVS images must be able to access the primary Sysplex and CFRM (Coupling Facility Resource Management) couple data sets. For more information on couple data sets and coupling facilities, see Chapter 2, “Sysplex functions” on page 9.

msys for Operations must be installed on every system of the Parallel Sysplex. Since NVSS, and therefore *msys for Operations*, is a VTAM application, *msys for Operations* must be defined to VTAM during the installation (see “Preparing VTAM” on page 32). The individual instances of *msys for Operations* use XCF and XES for mutual communication (through an XCF group) and for triggering maintenance and recovery actions.

Tasks and message automation

This section provides a short introduction to some key concepts of the NVSS part of *msys for Operations*.

Understanding tasks

msys for Operations is internally organized by *tasks*. It is helpful for system programmers to have a basic understanding of this internal structure because they will have to specify names and attributes for certain tasks during the installation.

A task is a process with certain properties that is defined to, and runs within, msys for Operations. Tasks can be started and stopped. Every request you put to msys for Operations is executed by a task.

One common type of task are *operator station tasks* (OSTs). An OST is started when you have logged on to msys for Operations successfully. The OST establishes and maintains the online session with you. It receives your commands, executes them, and displays the messages that are sent in response. The OST will be stopped again when you log off. OSTs are defined to msys for Operations by the operator ID and a password, and by a set of authorizations.

OSTs serve to receive and execute the commands of a human operator. Now, the automation procedures of msys for Operations also need tasks that execute commands for them. For these, OSTs that were started by an operator logon are not suitable. First, one cannot be sure that the respective user is really logged on, and second, an automation procedure needs no terminal to pass a command to the task that is to execute it.

To accommodate the needs of automation, msys for Operations provides another type of task that is called *autotask*. Autotasks perform functions similar to those of OSTs, but they do not require a terminal or online user. They are usually started immediately after the start of msys for Operations and are active until msys for Operations is shut down. One important application of autotasks is in connection with message automation. This will be described in the following section.

Note that autotasks and OSTs do not differ in their definition. Both are defined by an ID and certain authorizations. The difference lies in the way in which they are started. OSTs are started by an operator logon. Autotasks are started by a command. Thus, the same task can be started as an OST and as an autotask. For the definition of IDs for human operators and autotasks, see “Defining operators, passwords, and logon attributes” on page 49.

OSTs and autotasks are not the only task types. An important task that is neither an OST nor an autotask is the automation router task. The automation router task receives the automatable MVS messages and plays a crucial role in message automation (see “Automatic recovery and message automation”). Its name must be specified during the installation of msys for Operations (see “Customizing the initialization style sheet” on page 34).

Automatic recovery and message automation

msys for Operations performs most of its automatic recovery actions in response to certain messages that are issued by the sysplex components of MVS (XCF and XES). In broad outline, this message automation works as follows.

msys for Operations supplies a hard coded message automation table that specifies for the relevant messages:

- What has to be done in response to the message
- Which task of msys for Operations is to execute the response

You must define these messages as automatable to the message processing facility (MPF) of MVS during the installation of msys for Operations. For details refer to “Updating member MPFLSTxx” on page 28. If appropriate definitions exist and MPF is activated, MPF forwards the messages to msys for Operations at runtime. In msys for Operations, the automation router task receives the message, reads the automation table, and routes the command that is associated with the message to the task that is associated with the command in the table. The associated task will then issue the command.

The tasks that perform the automated recovery actions have been started as autotasks immediately after the start of msys for Operations and will be active as long as msys for Operations itself is running. In this way, recovery actions can be performed even when no human operator is logged on to msys for Operations. In order to enhance performance, msys for Operations distributes the automated responses to several autotasks that work in parallel.

For example 1 on page 3, the message automation works as follows:

1. When the alternate CDS for a certain CDS type has been made the primary, XCF issues the message IXC263I, which implies that an alternate CDS is no longer available (for details on CDSs, see “Managing couple data sets” on page 9).
2. If IXC263I has been defined as automatable, the message is forwarded to msys for Operations. The automation router task receives the message.
3. The automation router reads the automation table. The automation table contains an entry for IXC263I. The command specified in that entry will create a new alternate CDS. The entry also specifies the task that is to execute that command. The command is routed to the task that is specified in the table entry.
4. The target task executes the command.
5. A new alternate CDS is created for the CDS type in question. This includes:
 - a. Allocating the new CDS by calling the XCF formatting tool
 - b. Defining the formatted data set to XCF

Figure 2 on page 8 illustrates the information flow for this example:

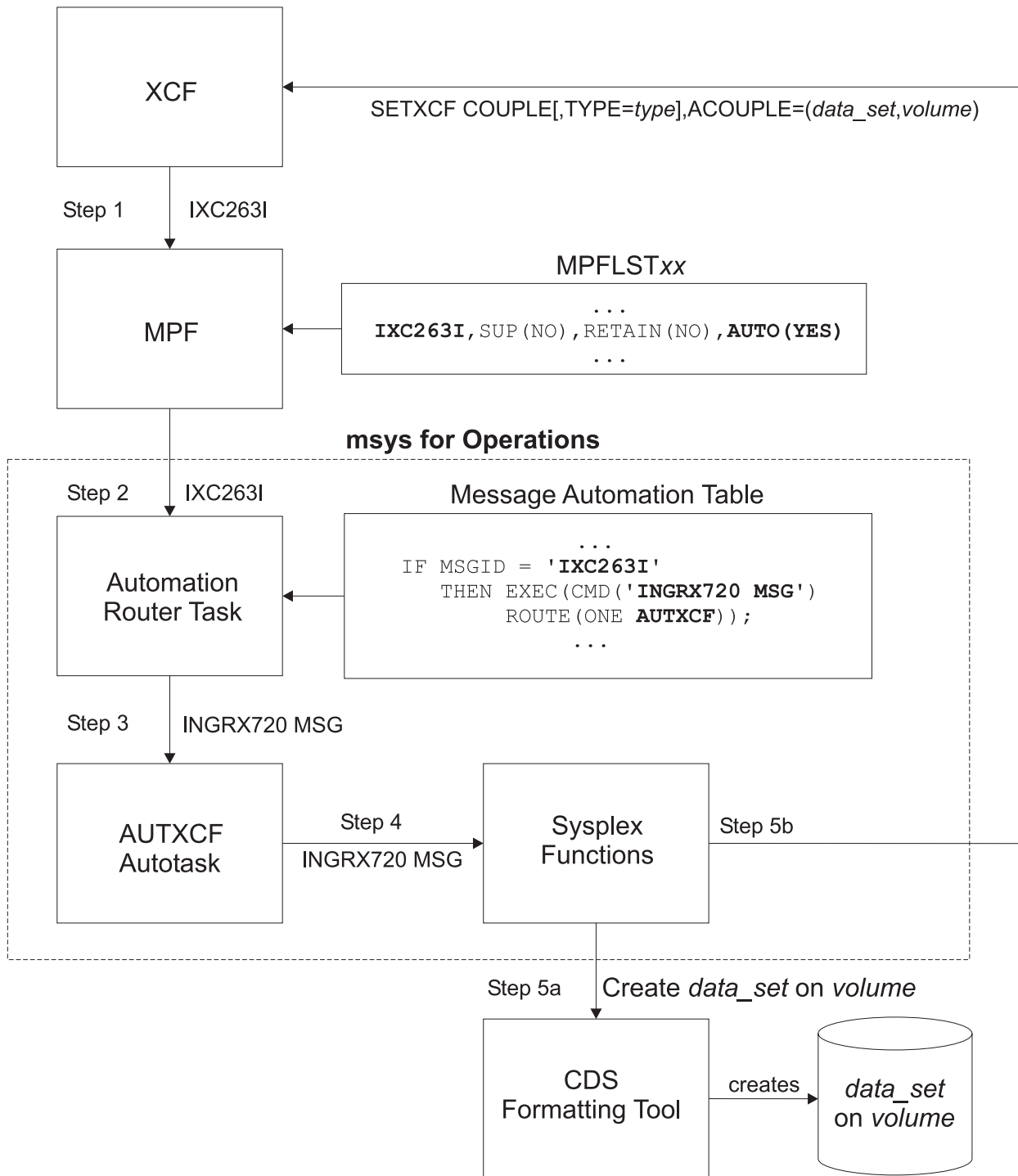


Figure 2. Automation for message IXC263I

Chapter 2. Sysplex functions

This section gives an overview of the sysplex functions that are available with msys for Operations. The functions are grouped by sysplex components, for example, couple data sets, coupling facilities. Every subsection describes the functions of one group and indicates how these functions can be customized. All customization information must be defined in the member AOFCUST, which is described in detail in “Editing the customization member AOFCUST” on page 35.

At the beginning of each section, the relevant sysplex concepts are briefly explained. These explanations only provide some basic information. For a full understanding of a Parallel Sysplex, refer to *z/OS MVS Setting Up a Sysplex* and *z/OS MVS Programming: Sysplex Services Guide*.

The emphasis is on the actions that have an impact on the sysplex configuration. Display functions are only mentioned.

Managing couple data sets

Couple data sets (CDSs) contain control information about the sysplex and its resources. They are of crucial importance for the functioning of a Parallel Sysplex. This holds especially for the SYSPLEX couple data set, which contains information about the systems and the communication structure (XCF groups) of the sysplex, and for the CFRM couple data set, which specifies its coupling facilities (CFs) and structures (see “Managing coupling facilities (INGCF command)” on page 14). Every MVS system of a Parallel Sysplex must have access to these CDSs, as well as to those of all other implemented sysplex functions (for example, SFM and ARM).

When a member system cannot access a CDS the corresponding sysplex function is impacted, and in some cases the sysplex will go down. Therefore, it is recommended to define two CDSs to XCF for every CDS type required by the implementation of the sysplex. One of these, the *primary* CDS, is the one that is actually used. The other one, which is called the *alternate* CDA, serves as a backup copy. The two CDSs contain the same data. Whenever the primary CDS changes, XCF updates the alternate CDS accordingly. If an alternate CDS is available for a certain type, XCF automatically switches to this alternate CDS whenever a member can no longer access the primary CDS.

All CDSs except the sysplex couple data set contain one or more user-defined configurations, called *policies*. For each CDS type, only one policy can be active at any point in time. However, it is possible to switch the active policy at runtime. Refer to “INGPLEX CDS” on page 10 for further information.

msys for Operations offers two functions for easier CDS management:

- Automated creation and recovery of alternate couple data sets for continuous availability
- INGPLEX CDS, which simplifies management of couple data sets

Ensuring continuous availability of alternate couple data sets

When an alternate CDS exists for a given CDS type and the current primary CDS fails, XCF makes this alternate the primary CDS. After this switch, however, an alternate CDS no longer exists, and if the former alternate and current primary

CDS also fails, the problems that were to be avoided by the creation of an alternate arise again. To exclude this single-point-of-failure situation, msys for Operations provides a recovery mechanism that tries to ensure that an alternate CDS is always available for every CDS type used.

msys for Operations creates a missing alternate in the following two situations:

- At initialization time msys for Operations checks if an alternate is specified for every existing primary CDS. If there is a primary CDS for which no alternate exists, msys for Operations automatically creates the missing alternate.
- At runtime, msys for Operations ensures that a new alternate is created whenever the current alternate has been removed or made the primary.

INGPLEX CDS

INGPLEX CDS displays information about all couple data sets, including details of the respective policies, and allows the operator to perform the following actions for every CDS type that is required by the implementation.:

- Switch from the primary to the alternate CDS
- Define a new alternate CDS
- Change the active policy (if applicable)
- Automatically rebuild a structure after the switch

For the first two actions, INGPLEX CDS offers automatic creation of a new alternate CDS. You can also specify your own alternate CDS. For more information on INGPLEX CDS, see “CDS” on page 128.

Customization aspects

Recovery of alternate CDSs is initiated either by the CDS function of INGPLEX or in the background (for example, at initialization time). Background recovery can be switched on and off by setting or unsetting CDS in the AUTO section of AOFCUST. Automatic recreation with INGPLEX CDS is always enabled. It cannot be switched off in the AUTO section.

You must specify the spare volumes that msys for Operations may use for recreating missing alternate CDSs (CDS section of AOFCUST). This is also required for automatic recreation with INGPLEX CDS. Every CDS type has its own pool of spare volumes. Note that if you define no spare volumes for a CDS type, no recovery will be performed for this type.

For details on the AOFCUST entries, see “Editing the customization member AOFCUST” on page 35.

You can control access to those functions of INGPLEX CDS that modify the sysplex configuration. Refer to “Command authorization” on page 51 for details.

Managing the system logger

The system logger provides a sysplex-wide logging facility. Applications that use the system logger write their log data into *log streams*. Within a Parallel Sysplex, these log streams are usually associated with a coupling facility structure. For further information about coupling facility structures, refer to “Managing coupling facilities (INGCF command)” on page 14. By using a coupling facility log stream, members of a multisystem application can merge their logs even when residing on different systems.

When an application writes data to a log stream this data is stored at first temporarily in the associated structure (coupling facility log stream) or a local buffer (DASD-only log stream). From there, it is offloaded into a log stream data set which is automatically allocated by the system logger. When this log stream data set is full, the system logger allocates a second one, and so on.

The control information for the system logger, which includes a kind of directory for the log stream data sets of every log stream, is contained in the LOGR couple data set. The total number of log stream data sets that can be allocated by the system logger is determined when the LOGR couple data set is formatted.

Two problems that can arise in connection with the log stream data sets are a shortage of directory space in the LOGR CDS and incorrect share options for the log stream data sets. msys for Operations provides the following recovery actions for these problems:

- The primary and alternate LOGR CDS are automatically resized in case of a directory shortage
- The operator is notified when the share options for log stream data sets are not defined correctly

Resizing the LOGR couple data sets in case of directory shortage

The LOGR CDS contains, among other things, information about the log stream data sets used by the system logger. This information is stored in *directory extents*. Every directory extent record can hold information about up to 168 log stream data sets. The number of directory extents available in a LOGR CDS is specified when the CDS is formatted (DSEXTENT parameter). When all available directory extents have been used up, the system logger can no longer allocate new log stream data sets. This can raise considerable problems for applications that use the system logger.

With msys for Operations, you can avoid this situation. If you switch on logger recovery, msys for Operations automatically reformats your primary and alternate LOGR CDS with an increased DSEXTENT parameter whenever the system reports a directory shortage.

Notifying the operator of incorrect share options

If you want to use the system logger, you must define share options for the log stream data sets. Merging data from several systems into one coupling facility log stream requires that you specify VSAM SHAREOPTIONS(3,3) for the log stream data sets. With other share options, especially (1,3), such a merge will fail. If you manage your DASD data sets with SMS (Storage Management Subsystem), a possible cause for incorrect share options is that the data class you use for the log stream data sets is also used for other purposes that require different share options.

msys for Operations provides a control mechanism for VSAM share options. The share options are checked on a daily basis. If incorrect share options are detected, msys for Operations notifies the operator.

Customization aspects

Automation of system logger recovery is enabled by setting LOGGER in the AUTO section of AOFCUST.

No further customization is required. Note however that if you switch on automation for system logger recovery, you also activate the recovery function for alternate CDSs (see “Ensuring continuous availability of alternate couple data sets” on page 9), even if CDS automation is switched off. For details, see “Editing the customization member AOFCUST” on page 35.

Recovery functions

Resolving a system log failure

When the system log is inactive, msys for Operations tries to start it by issuing the WRITELOG START command. In addition, it tries to make it the hardcopy log.

The SYSLOG message automation has been enhanced by a recovery function. Both functions (recovery and automation of message IEE043I) exist in parallel. The recovery takes place in case the system log becomes inactive. It responds to message IEE037D following one of the messages IEE043I, IEE533E, and IEE769E. In addition, it responds to message IEE041I. The remaining messages are just trapped and saved for the recovery later on. Except for the decision message you can define individual action commands in the customization dialogs for the above messages.

Because the recovery and the old automation of message IEE043I affect the same resource SYSLOG, only one threshold can be defined under the policy SYSLOG THRESHOLDS. To allow the controlling of the SYSLOG recovery separately from the old SYSLOG message automation, the new minor resource flag LOG is introduced. For the run-time environment, two thresholds are generated from the single threshold definition. The names of thresholds correspond to the names of the minor resource flags.

Note: Action commands that are executed for the old SYSLOG message automation are defined in the customization dialog using entry SYSLOG in the messages policy for entry type MVS Components. Action commands that are executed for the new SYSLOG recovery of message IEE043I are defined in the customization dialog under entry “IEE043I” in the same policy. In case SYSLOG message and recovery commands are defined, both action commands will be issued if message IEE043I followed by message IEE037D is trapped.

Customization aspects

Automation of system log recovery is enabled by setting LOG in the AUTO section of AOFCUST.

Resolving WTO(R) buffer shortages

When all WTO(R) buffers are used up, it can happen that commands can no longer be processed. To resolve this situation, there are several options. You can extend the buffer, change the properties of the affected consoles, or cancel jobs that issue WTO(R)s.

msys for Operations provides recovery of buffer shortage in two stages. First it tries to extend the buffer and to modify the console characteristics, if applicable. If this does not help, it cancels jobs that issue WTO(R)s. The jobs that can be cancelled by msys for Operations in case of a buffer shortage must be specified by the user.

Customization aspects

Automation of buffer shortage recovery is enabled by setting WTO in the AUTO section of AOFCUST.

If you want msys for Operations to cancel jobs issuing WTO(R)s in order to resolve a buffer shortage, you must specify these jobs in the WTOBUF section of the AOFCUST member.

For details on the AOFCUST entries, see “Editing the customization member AOFCUST” on page 35.

Handling Long-Running Enqueues (ENQs)

This new recovery function lets you:

- Check which resources are blocked
- Customize the automation to cancel or keep the jobs that block the resource
- Customize the automation to dump the jobs before they are cancelled

You can determine which resources you want to monitor. You can define a value for the maximum time a job can lock a resource while other jobs are waiting for the resource. If the amount of time is exceeded, the recovery takes place.

Identifying and eliminating these potential bottle-necks helps to reduce the risk of a parallel sysplex outage.

Resolving Pending I/Os for Systems Being Removed From the Sysplex

This new function helps to avoid sysplex outages by identifying pending I/Os that can block the system. Under certain circumstances, SFM cannot complete the isolation of a failed system. This is because SFM’s HW isolation, resetting the channel subsystem (CSS) of the failed system, is driven through the CF. When connectivity between the system image and the coupling facility is lost, SFM cannot perform the hardware isolation (ISOLATE command) and defers resetting the system image until manual operator intervention occurs. Message IXC102A tells the operator to manually reset the HW and then reply “down” to the message, after which SFM safely partitions the system image out of the sysplex. The longer the delay lasts the more the components and applications that rely on XCF messaging are impacted. The delay can eventually lead to a sysplex outage when the failed system has I/O operations pending. The automation of the message IXC102A minimizes the delay.

The first part of the automation clears any pending I/O operations by sending a hardware command to the Support Element. This requires information about the software running on the hardware. Because the system issuing the message IXC102A does not necessarily have access to the hardware of the failed system, the automation needs the predefined mapping between software and hardware. Depending on this mapping it then routes the hardware command to the system having access to the hardware of the failed system. For further information about the hardware requirements refer to “Preparing the Hardware” on page 23.

Recovering Auxiliary Storage Shortage

With the automation of local page data sets msys for Operations prevents auxiliary storage shortage outages by dynamically allocating spare local page data sets when needed. The new function:

- Checks which jobs cause the shortage condition

- Checks whether additional page data sets can be added. If this is not possible, jobs can be cancelled.

To enable local page data set automation customize the PAGTOTL parameter (defined in one of the IEASYSxx PARMLIB members used during IPL). Make sure to set the PAGTOTL parameter to a value greater than the number of local page data sets currently used.

Local page data sets must be defined in the master catalog and should not be SMS-managed. It is recommended to use preallocated local data sets instead of dynamically allocated ones. This makes the process faster because formatting newly allocated page data sets is timeconsuming (10sec./35MB). Each predefined local page data set should be allocated with 10% space of local page space currently used by the system. If predefined page data sets can no longer be allocated, new local page data sets will be created dynamically.

Managing coupling facilities (INGCF command)

A *coupling facility* (CF) is a logical partition that provides storage for data exchange between components of an application that is distributed across different systems in a Parallel Sysplex. A Parallel Sysplex can contain more than one CF. The storage of a coupling facility is divided into areas that are called *structures*. You can imagine a structure as a special kind of data set. It is these structures, which are identified by their name, that are accessed for reading and writing by the application components.

The association between CFs and structures is dynamic. A structure that is used by an application need not be allocated at all (for example, when the application is not running), and can be allocated on different CFs at different points in time. For every structure, there exists a *preference list* that defines the CFs on which it may be allocated. The order of the CFs in that list determines which CF is selected when more than one member of the list satisfies all allocation requirements (for example, provides enough space).

The preference list, the space requirements, and other properties of the structures are defined in the active CFRM policy. This policy is contained in the CFRM couple data set. Refer to “Managing couple data sets” on page 9 for further information.

XES allocates a structure that does not yet reside on any CF when an application component requires to be connected to it. Note that the application component only specifies the name of the structure that it wants to access. It is up to XES to decide on which CF the structure is allocated. This decision is mainly influenced by the structure definition in the active CFRM policy. After the structure has been allocated, the requesting application component can access it, and further components of this application can require to connect to it. An application component that has access to an allocated structure is referred to as an *active connector* to this structure.

In the simplest case, XES deallocates a structure when all connected application components have disconnected from the structure. However, an application component can require that the structure or its own connection to the structure be *persistent*. When the *structure* is persistent it remains allocated even when no application component is connected to it any more. When a *connection* is persistent the structure remains allocated after a failure of that connection. The application component in question remains a connector to the structure, although not an active

one. It is now a *failed persistent* connector. In both cases, you can force the deallocation of the structure as soon as it has no more active connectors.

Allocated structures can be *rebuilt*. Rebuild is the process of reconstructing a structure on the same or another CF. A rebuild consists of three main steps. First, XES allocates the new structure instance. Then, the data of the old structure is reconstructed in the new structure. Finally, XES deallocates the old structure instance. Note that you cannot specify the target CF in your rebuild request. As with the structure allocation, XES selects it from the preference list.

There are two methods for rebuild, user-managed and (from OS/390 2.8 onward) system-managed. With user-managed rebuild, the active connectors are responsible for reconstructing the data. With system-managed rebuild, XES transfers the data to the new structure instance. Therefore, system-managed rebuild is also available for structures without active connectors. These structures can either be themselves persistent or have failed persistent connections.

When an application component connects to a structure, it specifies whether it allows the structure to be rebuilt through user-managed or system-managed rebuild. For structures with active connectors, both rebuild methods require that all active connectors have allowed the respective rebuild method.

You can also *duplex* structures. Duplexing means maintaining two instances of the same structure on different CFs at the same time. Duplexing serves to increase availability and usability of a structure.

Typical management tasks for CFs are removing a CF from the sysplex and reintegrating it again. These tasks consist of several steps that must be performed in a certain order and can be rather complex in themselves. To simplify these operations, msys for Operations offers the INGCF command. INGCF has several functions, which serve to manipulate structures and the CFs themselves. These functions are briefly described in the following. For more information, see “INGCF” on page 106.

DRAIN

INGCF DRAIN displays information about the allocated structures of a CF and supports removal of this CF from the sysplex. Usually, draining a CF requires that at least one alternate CF be enabled for the sysplex.

With INGCF DRAIN, you can perform the following sequence of tasks:

1. Rebuild all structures that can be rebuilt by user-managed or system-managed rebuild on an alternative coupling facility, and deallocate structure instances on the target CF that are being duplexed on another CF.

The scope of the rebuild action depends, among other things, on the release level of the systems from which the structures were allocated:

- Structures that were allocated from a system with OS/390 2.7 or below can only be rebuilt when they have at least one active connector and all its active connectors support user-managed rebuild
- Structures that were allocated from a system with OS/390 2.8 or above can be rebuilt when they have an active connector and support either user-managed or system-managed rebuild, or when they have no active connector.

Note: INGCF DRAIN rebuilds structures one at a time (SETXCF START,REBUILD,STRNAME=), not globally (SETXCF START,REBUILD,CFNAME=), and always on a CF that is different from the target CF (LOCATION=OTHER).

2. Force the deallocation of structures that have no active connectors and could not be rebuilt.
3. Disconnect the coupling facility from the systems with which it is connected.
4. Inactivate the coupling facility.

INGCF DRAIN ensures that the supported actions are carried out in the right order. Thus, for example, INGCF DRAIN lets you disconnect the coupling facility from the systems only after all structures of the coupling facility have been moved to another CF or have been deallocated. After each step, INGCF DRAIN presents the results of that step. You can then choose whether you want to initiate the next step.

ENABLE

INGCF ENABLE is the counterpart of INGCF DRAIN. It supports integration of a new CF into a sysplex and reintegration of an existing CF into the sysplex, for example, after maintenance of the CF.

Note: INGCF ENABLE assumes that the receiver paths from the CF to the systems of the sysplex have been defined and activated. This requires a POR of the CPC on which the CF resides.

With INGCF ENABLE, you can perform the following sequence of tasks:

1. Activate the coupling facility
2. Connect the systems of the sysplex with the coupling facility (sender paths)
3. Switch to another CFRM policy if
 - The target CF is not defined in the active policy, and
 - A policy is available that contains the target CF and definitions for all active CFs and all allocated structures
4. Populate the target CF, that is, rebuild all those structures on the target CF whose preference list starts with this CF, provided that this is not excluded by other requirements

When the structures have been allocated on the target CF, INGCF ENABLE displays the result.

As INGCF DRAIN, INGCF ENABLE ensures that the supported actions are carried out in the right order. Thus, you can only start populating the target CF after it has been connected to the systems of the sysplex.

PATH

IINGCF PATH lets you set the sender paths ONLINE or OFFLINE.

STRUCTURE

INGCF STRUCTURE displays all the allocated structures of a CF and information about their actual conditions. For a selected structure, you can:

- Display detail information
- Initiate a rebuild on another CF depending on the rebuild pending status (PENDING calls location=NORMAL, otherwise location=OTHER)
- Force the deletion of the structure

- Starts and stops duplexing

Rebuild and deletion can only be performed for structures with certain conditions.

Customization

None. For information on how to control access to INGCF, refer to “Command authorization” on page 51.

Note that the ENABLE function requires that the active IODF is catalogued. Otherwise, sender path information cannot be retrieved in certain situations.

Miscellaneous

Recording IPL Information

With the INGPLEX IPL command you can record, view and compare the IPL information of the operating system. If a system does not behave after the IPL as expected, the IPL recording function enables you to identify parameters that were changed, for example, since the last IPL. The recording function enables you to compare different IPL scenarios. INGPLEX IPL is a tool that helps to identify and resolve the cause of startup problems. For further information about the INGPLEX IPL command refer to Chapter 14, “Sysplex-related commands” on page 105.

Dump options

The enhanced INGPLEX functions allow to specify dump options, and provides an easy-to-use interface for multisystem dumps. Furthermore the new INGPLEX functions let you view, enable, disable, and delete SLIP traps defined in the sysplex. For further information refer to Chapter 14, “Sysplex-related commands” on page 105.

Part 2. Setting up msys for Operations

This part describes how to install and customize msys for Operations.

Chapter 3. Installing msys for Operations

Functional Prerequisites

To obtain current service recommendations and to identify current product service requirements, contact the IBM Customer Support Center or use S/390 SoftwareXcel to obtain the current "PSP Bucket".

The following lists the functional hardware and software prerequisites that are required for using the enhanced Parallel Sysplex automation functionality shipped with APAR OW50146.

Software Prerequisites

The following APARs need to be installed:

APAR	PTF	FMID	Area	Function
OW51923	UW84343	HBB7703	SPI	BCP (Basic Control Program) internal interface used by system recovery and coupling facility functions
	UW84344	HBB7705		
	UW84345	HBB7706		

This APAR allows for the simultaneous usage of the BCP internal interface by multiple applications, especially by HCD and msys for Operations. msys for Operations uses this interface for activating, inactivating, and querying a coupling facility as well as for sending a hardware command to the LPAR of the system which is being partitioned out of the sysplex.

APAR	PTF	FMID	Area	Function
OW52369	UW86446	HBB7703	XCF	Coupling facility functions
	UW86444	HBB7705		
	UW86445	HBB7706		

With z/OS 1.2 XCF introduced a different behavior from previous functionality of how users are informed when a rebuild duplex process has completed. Only users starting the duplex process are informed when this process has been stopped. Therefore, all automation functions doing rebuilds will time-out for structures whose duplex process has been started by MVS or by another operator. This APAR resolves the situation by informing the user who started the duplex process as well as the user who stops it.

APAR	PTF	FMID	Area	Function
OW53529		HBB7703	GRS	Elimination of long running ENQs
		HBB7705		
		HBB7706		

The automation of eliminating long-running ENQs is only applicable when this APAR is installed. After installing the APAR you can activate the automation by issuing the following command from the NCCF screen of the NVSS where you want to run this sysplex-wide automation: The ACF COLD command will schedule the automation function on the appropriate autotask when the APAR has been applied. Recycling the NVSS address space has the same result.

APAR	PTF	FMID	Area	Function
OW53637		HBB7703	XCF	Structure rebuild function
		HBB7705		
		HBB7706		

With z/OS 1.2 XCF introduced a different behavior from previous functionality of how users are informed when a rebuild duplex process has completed. This also affects messages issued on behalf of the "SETXCF ALTER" command. When the automation rebuilds a structure, it also checks its initial and actual size. In case the initial size is less than the actual size the automation tries to change the initial size to the actual size. Because the expected message is no longer issued as a command response the automation times out when waiting for it and issues an appropriate message. Even if automation continues the message could be misleading. This APAR resolves the problem.

Hardware Prerequisites

Required Support Element LIC Levels

For the current information about the required LIC levels for the following servers refer to the "PSP Bucket".

- zSeries
- CMOS-S/390 G5,G6
- CMOS-S/390 G3,G4

Required Hardware Management Console LIC Level

- Driver 3cg

This MCL level is required for all HMCs that serve as Master HMCs and have the LIC change console service enabled. Note, that at least one HMC in your processor LAN configuration must have this service enabled in order to provide cross-CPC-communication over the BCP internal interface.

The following sections describe how to install and customize msys for Operations. It assumes that you have completed all the steps that are described in the program directory, and that the target libraries for both the Sysplex Functions and the NVSS part of msys for Operations are available.

When you are using SA OS/390 V1R3 and migrate from OS/390 or z/OS V1R1 to z/OS V1R2 you can enable msys for Operations functions and run SA OS/390 V1R3 on the same systems. For details, see Appendix C, "Coexistence of msys for Operations and SA OS/390 V1R3" on page 227. When you are using SA OS/390 V2R1, you must install PTFs UW90711 and UW90712 (APARs OW37539 and OW48837). This provides all the functions of msys for Operations within SA OS/390 V2R1, so that there is no need to install msys for Operations.

The installation of msys for Operations consists of the following steps:

- Preparing the hardware
- Planning for domain and console names
- Preparing the MVS System
- Preparing the data sets of msys for Operations
- Defining msys for Operations to VTAM
- Updating the start procedure and the initialization style sheet
- Editing the AOFCUST member by which the Sysplex Functions are customized

msys for Operations must be installed and run on every MVS image of the Parallel Sysplex you want to control or retrieve information from. The customization effort can be minimized by sharing the data sets in which msys for Operations is

installed and through the use of symbols where systems require certain unique values. This can be done by using system symbols that are resolved differently on every system.

Preparing the Hardware

This section describes the steps necessary to prepare your processor hardware in the sysplex to use the Parallel Sysplex enhancements shipped with APAR OW50146.

Understanding the Hardware Interface

In order to allow the sysplex-wide activation or deactivation of the coupling facilities and to control sysplex members leaving the sysplex, msys for Operations uses the BCP (Basic Control Program) internal interface. The BCP internal interface of the following processor hardware families is supported:

- zSeries
- CMOS-S/390 G6
- CMOS-S/390 G5

Using the BCP internal interface from MVS, allows to send hardware operations commands such as SYSTEM RESET or ACTIVATE to the Support Element attached to the own processor hardware (CPC). If the CPC is configured in LPAR mode, the operations command can be sent to all logical partitions defined on the CPC.

Furthermore, with the enhanced sysplex functions of msys for Operations, sysplex members running on other CPCs than the own image can be controlled through the BCP internal interface. This is possible by defining all CPCs of your sysplex to the master HMC of your processor hardware LAN.

The following processor hardware can be controlled as a target with the BCP internal interface of the above listed processors, but cannot use the msys for Operations BCP internal interface to control itself or other processors:

- CMOS-S/390 G4
- CMOS-S/390 G3

Note, that the MVS/HCD function uses the BCP internal interface to update IOCDS and IPL information in the Support Elements of addressed CPCs. You cannot use msys for Operations to perform these tasks, nor can HCD be used to perform the hardware operations functions of msys for Operations.

Preparing the Master Hardware Management Console

Log on to the HMC in your LAN to be used for change management operations with a user ID having SYSPROG authority. The HMC must have the CPC objects of your sysplex in the defined CPCs group.

Select **Console Actions** and click on the *Enable Hardware Management Console Services* icon. Set the LIC change Enabled radio button. Press the OK button to save the change or press cancel if LIC change was already set to Enabled.

Usually, there is one HMC in a CPC LAN environment, that has LIC change permanently enabled. It will automatically be used by the BCP internal interface. Make sure this HMC has all CPC objects of your sysplex in the Defined CPCs group.

Preparing the Support Element

Before the BCP internal interface can be used, the CPC Support Elements in your sysplex need to be verified, if the required prerequisite MCL levels are active and if the needed services are enabled with the necessary settings.

- Enable API and set the community name
- Cross partition flag (LPAR mode)
- Configure SNMP

Enabling the API and Set the Community Name

In order to communicate from one CPC with another CPC, the BCP internal interface needs the SNMP application API of the SEs enabled. For this task, you need to be logged on in *Access Administrator mode* on your CPC's support element.

Select **Console Actions** and click on the *Support Element Settings* icon. On the Support Element Settings Notepad, choose the *API* tab. If not already active, enable the Support Element Console application Program Interface by clicking on the enable button. Type in the community name you have chosen in the *Community name* field. Note, that only UPPERCASE letter community names are allowed when this HW interface is used.

Press the *Apply* button.

A message box is displayed informing you that you have to restart the Support Element in order to enable these changes.

Configuring SNMP

On the Support Element, the BCP internal interface uses a SNMP Manager application to communicate with the SNMP Agent of the Support Element. This application requires the *Community name*, previously defined with the Support Element Settings task, to be used. Additionally, the transport *protocol* for the SNMP requests associated with this community name and the SNMP Manager's *address* and *netmask* need to be defined. Finally, the *Access Type* that is allowed for SNMP requests from the Manager, has to be defined. Select **Console Actions** and click on the *SNMP Configuration* icon. On the SNMP Configuration Notepad, type the uppercase community name, you have chosen to use for the BCP internal interface, in the *name* field. From the *Protocol* selection list, choose UDP. It is recommended to specify the loopback address of the Support Element, 127.0.0.1 as the *address* and use this address also as the *Network mask*. This prevents SNMP requests of other Manager applications from being processed by the SE, only BCP internal interface requests can use this name.

As *Access Type* select the read/write radio button. This allows the BCP internal interface to modify object information in the Support Element. This is a required configuration step. For additional SNMP and API configuration information please refer to *zSeries 900 Application Programming Interface*, Chapter 6, Configuring the Data Exchange APIs.

Setting the Cross Partition Flags

For this task, you need to be logged on in *System Programmer mode* on your CPC's support element.

Click on the CPC Group and mark the CPC icon. Select the task **CPC Operation Customization**. Click on the *Change LPAR Security* icon. The displayed window is showing the security settings from the active IOCDS for the logical partitions defined on this CPC.

For each logical partition that should use the BCP internal interface in order to control another partition on this CPC, set the *Cross Partition Authority* Check Box.

Planning for the domain and EMCS console names

msys for Operations must be installed on every system of the sysplex. For each of these instances, you must specify a domain name, and that name must be unique across the network. Also, when you wish to run msys for Operations and a licensed NetView on the same system, this instance of NetView must have a domain name that is different from that of all the msys for Operations instances within the network.

Furthermore, the automation router task that receives MVS messages and passes them over to automation will be running within every instance of msys for Operations. For these tasks, you must define a name that is unique among all msys for Operations tasks across the sysplex. Moreover, when you run msys for Operations and a licensed NetView on the same system, the automation router task of the licensed product must also have a name that is different from those of all msys for Operations instances.

When planning a scheme for the domain and automation router names you might consider using system symbols for the following reasons:

- You can share certain data set members, for example, the msys for Operations start procedure (MSOAPROC), among all systems of the sysplex.
- You can add further systems to your sysplex, or install a licensed NetView on a system on which msys for Operations is running, without any changes to your naming conventions.

Domain names

Every instance of msys for Operations must be assigned a domain name that is unique across the network. The samples supplied with msys for Operations use MSO&SYSCONE as the domain name. &SYSCONE is a system-defined symbol whose default value is the last two characters of the system name, as specified for the &SYSNAME. symbol, which is also system-defined. You can override the default by specifying a value for &SYSCONE in IEASYMxx.

You can retain this name for all instances of msys for Operations if the following conditions are satisfied:

- The value of &SYSCONE is different for every system of the sysplex.
- You do not run an msys for Operations instance and a licensed NetView on the same system.

The domain name must be specified on one hand in the start procedure MSOAPROC of msys for Operations or the initialization style sheet (see “Preparing the startup procedure and the initialization style sheet” on page 33), and on the other hand in the VTAM definitions (see “Preparing VTAM” on page 32).

EMCS console names and the automation router task

Operator tasks use EMCS consoles to send commands from msys for Operations to the MVS operating system and to receive messages from MVS. The automation router task (see “Defining the name of the automation router task” on page 34) also uses an EMCS console to receive messages from MVS. You can explicitly assign an

EMCS console to an operator task or let msys for Operations assign it automatically. The EMCS console for the automation router task is always assigned by msys for Operations.

When an EMCS console is assigned to a task it is given a name. If you assign the console explicitly, you must specify that name. If msys for Operations assigns the console automatically, it uses the name of the task as the console name.

The important point is the EMCS console names must be unique across the sysplex, and that it is difficult to meet this requirement when the console names are identical with the task names. When, for example, operator A has logged on with the ID OPER1 on system KEY1, and has been assigned an EMCS console by msys for Operations, this console has the name OPER1. If operator B logs on subsequently on system KEY2 with the same operator ID, the automatic assignment of an EMCS console for the OST on KEY2 will fail, because a console with the name OPER1 already exists within the sysplex. B will not be able to issue MVS commands.

For operator tasks, you can avoid such a name clash by specifying the command list LOGPROF1 as the initial command in the operator profile (IC parameter in the NetView segment of RACF). This command list, which is supplied by msys for Operations, will then assign an EMCS console with a unique name to the operator task. For more details, see “Defining operator logon attributes in the NVSS segment of an SAF product” on page 50.

As for the automation router tasks of the individual msys for Operations instances, the only way to avoid naming conflicts of the associated EMCS consoles is to ensure that the task names are unique sysplex wide. The name of the automation router task for the individual msys for Operations instances is specified in the initialization style sheet. For details, see “Defining the name of the automation router task” on page 34.

Target libraries

Table 1 shows a list of target data sets as provided by the SMP/E installation process to be used for production on your system.

Table 1. Target data sets

Data Set Name	Description
NETVIEW.VnRnMn.BNJPNL1	NVSS panels
NETVIEW.VnRnMn.BNJPNL2	NVSS panels
NETVIEW.VnRnMn.BNJSRC1	not used
NETVIEW.VnRnMn.CNMCLST	NVSS clists
NETVIEW.VnRnMn.CNMINST	NVSS installation
NETVIEW.VnRnMn.CNMLINK	NVSS program library
NETVIEW.VnRnMn.CNMPNL1	NVSS online help
NETVIEW.VnRnMn.CNMSAMP	NVSS samples
NETVIEW.VnRnMn.DSIPARM	NVSS definition members
NETVIEW.VnRnMn.DSIPRF	NVSS operator profiles
NETVIEW.VnRnMn.NVULIB	NVSS program library
NETVIEW.VnRnMn.SCNMLNK1	NVSS link list modules

Table 1. Target data sets (continued)

Data Set Name	Description
NETVIEW.VnRnMn.SCNMLPA1	NVSS LPA modules
NETVIEW.VnRnMn.SCNMMAC1	NVSS macros
NETVIEW.VnRnMn.SDSIMSG1	NVSS messages
NETVIEW.VnRnMn.SDSIOPE1	NVSS key settings
NETVIEW.VnRnMn.SDUIMSG1	not used
NETVIEW.VnRnMn.SEKGCAS1	not used
NETVIEW.VnRnMn.SEKGLANG	not used
NETVIEW.VnRnMn.SEKGLNK1	not used
NETVIEW.VnRnMn.SEKGLUTB	not used
NETVIEW.VnRnMn.SEKGMOD1	not used
NETVIEW.VnRnMn.SEKGMOD2	not used
NETVIEW.VnRnMn.SEKGPNL1	not used
NETVIEW.VnRnMn.SKGSMP1	not used
NETVIEW.VnRnMn.SEZLCLST	not used
NETVIEW.VnRnMn.SEZLINST	not used
NETVIEW.VnRnMn.SEZLLINK	not used
NETVIEW.VnRnMn.SEZLPNLU	not used
NETVIEW.VnRnMn.SEZLSAMP	not used
NETVIEW.VnRnMn.SFLBDAT1	not used
ING.SINGMOD1	Sysplex Functions modules
ING.SINGMOD2	Sysplex Functions modules
ING.SINGMOD3	not used
ING.SINGNMSG	Sysplex Functions messages
ING.SINGNPNL	Sysplex Functions panels
ING.SINGNPRF	Sysplex Functions profiles
ING.SINGNPRM	Sysplex Functions DSIPARM and other samples
ING.SINGNREX	Sysplex Functions REXX execs
ING.SINGSAMP	Sysplex Functions general samples

Preparing the MVS system

This chapter describes the steps necessary to prepare your MVS system for the installation of msys for Operations.

Modifying the maximum number of language processor (REXX) environments for msys for Operations

Before the TSO/E language processor can process an exec, a language processor environment must exist. A language processor environment is the environment in which a REXX exec runs. The following discusses how msys for Operations uses these REXX environments and highlights issues to consider when estimating the number of language processor environments needed for your configuration.

msys for Operations provides several parts that contain REXX source code.

msys for Operations also contains several parts that make use of the Data REXX function. The Data REXX function allows the inclusion of REXX instructions and functions in data files.

When a REXX command list is run in msys for Operations, the REXX interpreter sets up a language processor environment for msys for Operations. When the command list ends, this unique environment can be held for reuse by the same task. msys for Operations retains these REXX environments to improve REXX environment initialization performance. As a result, it is very important to have a sufficient number of REXX environments available to msys for Operations. If more blocks are required than are available, msys for Operations issues the CNM416I REXX environment initialization error message.

msys for Operations retains up to three REXX environments and their associated storage until the operator logs off. Additionally, msys for Operations will *always* retain one REXX environment per task for Data REXX use.

The IRXANCHR table is a Time Sharing Option Extensions (TSO/E) table used to reserve storage for REXX environments. Both msys for Operations and TSO/E refer to this table when allocating storage for each REXX environment that is activated.

To change the number of environment table entries, you can use the IRXTSMPE sample that TSO/E provides in SYS1.SAMPLIB or you can create your own IRXANCHR load module. The IRXTSMPE sample is a System Modification Program/Extended (SMP/E) user modification (USERMOD) to change the number of language processor environments in an address space. The prolog of IRXTSMPE has instructions for using the sample job. The SMP/E code that is included in the IRXTSMPE sample handles the installation of the load module.

Note: IBM recommends that you set the number of environments to 200. This would require to set the ENTRYNUM parameter in the call of IRXANCHR call to 401.

Updating member SCHEDxx

Define the msys for Operations program as nonswappable in MVS. The msys for Operations program must run in MVS storage key 8.

To make the msys for Operations program nonswappable, use the SCHEDxx member of SYS1.PARMLIB. Ensure the SCHEDxx statement for the msys for Operations program is PGM=DSIMNT. You should use the sample INGSCH00 supplied with ING.SINGSAMP.

Updating member MPFLSTxx

In order to enable message automation with msys for Operations, you must code statements for the relevant messages in the PARMLIB member MPFLSTxx (see “Automatic recovery and message automation” on page 6). The INGEMPF member which resides in the Sysplex Functions sample library ING.SINGSAMP contains all the necessary statements. Copy these statements into your MPFLSTxx member.

AUTO(YES) is required in the .NO_ENTRY statement to gather all unknown WTORs. If you ensure that the unknown WTORs are routed to automation via the general

MPF exit IEAVMXIT and you have all messages that are specified in the message automation table also specified in MPFLSTxx with AUTO(YES), you can specify AUTO(NO) for the .NO_ENTRY statement.

Message Automation

Messages processed by the automation either via the NVSS MAT (message automation table) or by the NVSS commands TRAP and WAIT must not be suppressed by any MPF (message processing facility) list being used.

The following messages must be available for the Parallel Sysplex automation:

```
IEA230E IEA231A IEA232A IEA404A IEA405E IEA406I IEA794I
IEE037D IEE041I IEE043I IEE205I IEE400I IEE503I IEE533E IEE600I IEE712I IEE769E IEE889I
ILR009E
INGY1097I
IRA200E IRA201E IRA202I IRA204E
IXC247D IXC250I IXC251I IXC255I IXC500I IXC501A IXC559I IXC560A
IXG257I IXG261E
IXL126I IXL127A
```

Updating the link list and APF authorizations with PROGxx

The member INGPORG0 of ING.SINGSAMP contains a sample PROGxx member. Use this member to update the link list and the APF authorizations dynamically. In order to activate the updates, follow the instructions in the sample.

Note: The dynamic updates are only in effect until the next IPL. In order to make them persistent, add a line to your COMMNDxx member as follows:

```
COM='SET PROGxx'
```

Updating member COMMNDxx

Define the msys for Operations procedure collecting the IPL information in MVS. Add the following statement to a COMMNDxx Parmlib member that is shared by all systems in the sysplex:

```
COM='S HSAIPLC,SUB=MSTR'
```

Preparing msys for Operations

This chapter describes how to:

- Allocate the user data sets and VSAM clusters required by msys for Operations
- Load required members into user and system data sets

Allocating data sets and VSAM clusters using job INGALLC0

The job INGALLC0 is contained in member INGALLC0 of ING.SINGSAMP. It allocates partitioned data sets and VSAM clusters. They are listed, and their function is briefly explained, in Table 2 on page 30 and Table 3 on page 30. The data set and cluster names used in the tables are those that are supplied with the sample job; 'xxxxx' is taken as a place holder for the domain name. The third column specifies the DD statement of the msys for Operations startup procedure MSOAPROC where the data set or cluster is to be specified (see "Modifying the msys for Operations startup procedure" on page 33).

The PDSs are listed in the following table.

Table 2. Partitioned data sets for msys for Operations

Data Set Name	Function	DD Statement
MSOPS.USER.DSIPARM	Will hold shared definition members, for example the AOFUCST (see “Editing the customization member AOFUCST” on page 35).	DSIPARM
MSOPS.USER.xxxxx.DSIPARM	Will hold system-specific definition members.	DSIPARM
MSOPS.USER.xxxxx.DSILIST	See note 1.	DSILIST
MSOPS.USER.xxxxx.DSIASRC	See note 1.	DSIASRC
MSOPS.USER.xxxxx.DSIARPT	See note 1.	DSIARPT
MSOPS.USER.xxxxx.VTAMLST	Will contain the VTAM source definitions. See “Preparing VTAM” on page 32.	DSIVTAM
MSOPS.USER.VTAMLIB	Will contain VTAM load modules. See note 2.	
Notes: 1. This data set is not required for msys for Operations. However, this data set is required for NetView operation. It is strongly recommended that you allocate this data set to enable installation of the NetView product in the future. 2. Add the VTAMLIB data set to the list of authorized libraries in the IEAAPFxx of SYS1.PARMLIB.		

The following table contains the VSAM clusters:

Table 3. VSAM clusters for msys for Operations

Data Set Name	Function	DD Statement
NETVIEW.VnRnMn.xxxxx.DSILOGP	Contains the primary network log. See Chapter 11, “Using the netlog” on page 85.	DSILOGP
NETVIEW.VnRnMn.xxxxx.DSILOGS	Contains the secondary network log. See Chapter 11, “Using the netlog” on page 85.	DSILOGS
NETVIEW.VnRnMn.xxxxx.DSITRCP	Contains the primary trace log.	DSITRCP
NETVIEW.VnRnMn.xxxxx.DSITRCS	Contains the secondary trace log.	DSITRCS
NETVIEW.VnRnMn.xxxxx.DSISVRT	Saves internal information for an eventual restart of msys for Operations.	DSISVRT
ING.xxxxx.STATS	Contains the automation status file. AOFSTAT This file is used by the Sysplex Functions of msys for Operations.	AOFSTAT
ING.xxxxx.IPLDATA	Contains IPL information recorded after IPL. This file is used by the Sysplex Functions of msys for Operations.	HSAIPL

Note: The records needed for the system to view databases as active data sets are added during msys for Operations initialization.

Edit the sample job as described in the comments of the sample, and submit the job.

Loading members of partitioned data sets

This section documents data set members that you must copy from the target libraries to system or user libraries.

Loading DSIPARM members

Initially, the shared MSOPS.USER.DSIPARM data set and the system-specific MSOPS.USER.xxxxx.DSIPARM data sets (where xxxxx is the domain name) are empty. When performing the administration steps documented in this manual, copy the members you need to change from NETVIEW.VnRnMn.DSIPARM or ING.SINGSAMP to MSOPS.USER.DSIPARM or MSOPS.USER.xxxxx.DSIPARM. By doing this, you maintain a copy of the original member as it was installed from the distribution tape.

The edited members that can or should be kept in the shared MSOPS.USER.DSIPARM include:

- The customization member AOF Cust. For further information refer to “Editing the customization member AOF Cust” on page 35.
- The initialization style sheet CxxSTYLE. For further information refer to “Customizing the initialization style sheet” on page 34.
- The backup command authorization table CNMSBAK1 (to be copied from NETVIEW.VnRnMn.CNMSAMP). For further information refer to “Protecting immediate commands when CMDAUTH=SAF” on page 54.
- The specification member DSIDMNK for some system level parameters.

Adding Procedures to PROCLIB

Copy the following members of ING.SINGSAMP into a library that is part of the PROCLIB concatenation:

- INGNVAP0

This member contains a sample startup procedure MSOAPROC for msys for Operations. Rename the member to MSOAPROC. Review the procedure, at least for the high level qualifiers. See “Modifying the msys for Operations startup procedure” on page 33.

- INGPXCU, INGP HOM, INGP ILC, and HSAPIPLC

These procedures are used internally.

Note: INGPXCU and INGP HOM make use of certain data sets for which the started task users that are associated with these procedures and with NVSS must have the appropriate authorizations. For details, see “Granting NVSS and the STC-user access to data sets” on page 64.

The member CNMSJM04 in NETVIEW.VnRnMn.CNMSAMP contains a sample print job for the network log (see Chapter 11, “Using the netlog” on page 85) and the trace log. Copy this member to SYS1.PROCLIB and rename it to CNMPRT. If you defined passwords for the network log and the trace log (see “Defining Passwords for VSAM Databases” on page 71), add a password statement to CNMPRT.

Preparing VTAM

Because msys for Operations is a VTAM application, you must define it to VTAM. msys for Operations supplies a sample definition for an application major node in the member INGV TAM of ING.SINGSAMP. Copy this member to MSOPS.USER.xxxxx.VTAMLST, which was allocated by INGALLC0, and edit it if needed. When you use MSO&SYSCONE as your domain name, no changes are required.

Note: If you are using VTAM V4R4 or above, ensure that XCF signalling connectivity is in place in your sysplex, either with CTCs or with CF signalling structures.

Add this member to your VTAM configuration list ATCCON:xx. A sample list is contained in member CNMS0003 of NETVIEW.VnRnMn.CNMSAMP. You can copy this sample to MSOPS.USER.xxxxx.VTAMLST and edit it as required.

Modifying the application (APPL) major node

You may want to modify the following operands of the major node definition:

- Password
- Domain ID
- Logmode table

Changing the password (PRTCT parameter)

The original password on the ACBpassword keyword in CNMSTPWD (%INCLUDEd in the initialization style sheet) is MSO&SYSCONE. If you change this password, change *every* occurrence of PRTCT in INGV TAM to the same value. For example, if you change the password to PW006, then change:

```
MSO&SYSCONE. APPL AUTH=(NVPACE,ACQ,PASS),PRTCT=MSO&SYSCONE., X
```

to

```
MSO&SYSCONE. APPL AUTH=(NVPACE,ACQ,PASS),PRTCT=PW006,EAS=4, X
```

Changing the domain ID

When you change the domain ID from MSO&SYSCONE to something else, you must change *every* occurrence of MSO&SYSCONE in INGV TAM to the current domain ID, except for the password. For example, if you changed the domain ID to MYDOM, then you must also change the model definition

```
MSO&SYSCONE.* APPL AUTH=(NVPACE,SP0,ACQ,PASS),PRTCT=MSO&SYSCONE., X
```

to

```
MYDOM* APPL AUTH=(NVPACE,SP0,ACQ,PASS),PRTCT=MSO&SYSCONE., X
```

Notes:

1. An APPL name *prefixed* with MSO&SYSCONE cannot have its *suffix* changed. The new name must retain the LUC suffix. Thus, MSO&SYSCONE.LUC would have to be changed to MYDOMLUC.
2. If you code the optional ACBNAME operand on the APPL statement, it must match the APPL name in column 1. Therefore, if you change the domain ID, you must also change the value of ACBNAME.

Changing the logmode table (MODETAB parameter)

The member CNMS0001 of NETVIEW.VnRnMn.CNMSAMP contains a sample logmode table named AMODETAB that includes logmode entries for msys for

Operations sessions. The application definitions in INGV TAM point to entries in this table with their MOD ETAB and DLOGMOD parameters, for example:

```
MOD ETAB=AMOD ETAB,DLOGMOD=DSILGMOD
```

When you want to use this table, copy it to MSOPS.USER.xxxxx.VTAMLST (where xxxxx is the domain name) and edit it as required. Subsequently, assemble and link-edit it with Job CNMSJ006 (contained in NETVIEW.VnRnMn.CNMSAMP.CNMSJ006). CNMSJ006 places the assembled table into MSOPS.USER.VTAMLIB.

The definitions of AMOD ETAB take effect the next time you start VTAM or issue the VTAM command

```
MODIFY NET, TABLE, NEWTAB=AMOD ETAB, OPTION=LOAD
```

If you already have a table that contains all the required entries change the value of the MOD ETAB parameter to your table name, if necessary.

Preparing the startup procedure and the initialization style sheet

When you start msys for Operations, you use the START procedure MSOAPROC (INGNVAP0); you must review that procedure. Furthermore, you must customize the style sheet that contains the initialization information for msys for Operations.

Modifying the msys for Operations startup procedure

MSOAPROC (INGNVAP0) was copied to the PROCLIB when you loaded partitioned data sets during installation (see “Adding Procedures to PROCLIB” on page 31). Make the following changes to the msys for Operations startup procedure (MSOAPROC).

Setting symbols

You can set the following JCL symbols in MSOAPROC:

Q1

This symbol determines the high level qualifier for user data sets. In the sample procedure, it is set to ING.USER.

DOMAIN

With this symbol you can specify the domain name of the msys for Operations instance to be started. The domain must be defined to VTAM with this name. The name must be unique across the network. It must not contain more than five characters.

You have three options for specifying the domain name:

1. In the initialization style sheet
2. In the startup procedure
3. In the START command

The START command has the highest priority, the style sheet the lowest.

In the sample procedure, the DOMAIN symbol is set to MS0&SYSC LONE, where &SYSC LONE is a predefined system symbol. The default value of &SYSC LONE consists of the last two characters of the system name. However, you can override this value in IEASYMxx.

You can accept the settings of MSOAPROC, and share the startup procedure among all systems of the sysplex, if the following conditions are satisfied:

- The value of &SYSCclone is different for every system of the sysplex.
- You do not run an msys for Operations instance and a licensed NetView on the same system.

SQ1

This symbol determines the high level qualifier for the NVSS data sets. In the sample procedure, it is set to NETVIEW.VnRnMn.

SQ2

This symbol determines the high level qualifier for the Sysplex Functions data sets. In the sample procedure, it is set to ING.

VQ1

This symbol determines the high level qualifier for the VSAM clusters of NVSS. In the sample procedure, it is set to NETVIEW.VnRnMn.

NV2I

This symbol determines the name of the style sheet that is used for initialization of msys for Operations. It can also be used to construct unique names. The value must consist of exactly two characters. The default is 'NM'. In the sample procedure, no value is set.

The style sheet that msys for Operations is to use for initialization must have the name CxxSTYLE, where xx is the value of &NV2I. For example, if you set &NV2I to N1, msys for Operations will expect to find a member CN1STYLE in DSIPARM. If you do not specify a value for &NV2I, the member CNMSTYLE will be used.

If you wish to use the same style sheet for all msys for Operations instances of the sysplex, do not specify a value for &NV2I and keep CNMSTYLE in MSOPS.USER.DSIPARM.

Customizing the initialization style sheet

When msys for Operations is started it reads a member of DSIPARM that contains the initialization parameters. This initialization member is called *style sheet*.

The name of the style sheet that msys for Operations will access is controlled by the value of &NV2I in the msys for Operations start procedure. The value of &NV2I replaces the second and third characters of the default name CNMSTYLE. For example, a value of E1 for &NV2I causes msys for Operations initialization to use member CE1STYLE in DSIPARM. For more information on NV2I, see "Setting symbols" on page 33.

For initialization of msys for Operations, you must customize the SSIname entry and the TOWER entry.

Defining the name of the automation router task

The automation router task receives the unsolicited MVS messages that are defined as automatable in MPFLSTxx, and initiates the automated response if the automation table contains an entry for that message. You must specify the name of this task as the value of the SSIname keyword. It is essential that the name of this task is unique sysplex wide. This has the following reason:

The automation router task needs an EMCS console to receive MVS messages. The name of this console is identical with the task name. But EMCS consoles must have unique names across the sysplex. Therefore, the automation router task of every msys for Operations instance must have a different name from all other automation router tasks within the sysplex.

Provided that you do not run msys for Operations and a licensed NetView on the same system, you can ensure that the names are unique, and at the same time share the style sheet among the members of the sysplex, by defining

```
SSIname=&DOMAIN.SIR
```

where &DOMAIN has a different value on every system (for example, by setting the DOMAIN symbol to MS0&SYSCONE, see “Domain names” on page 25). Keep the shared style sheet in MSOPS.USER.DSIPARM.

Note: msys for Operations does not use the subsystem interface (SSI) of NetView, but the automation router task must be active nevertheless.

Activating the SA tower

In order to enable msys for Operations for the Sysplex Functions, remove the asterisk before 'SA' in the TOWER statement, that is, change:

```
TOWER = *SA *AON *MSM *Graphics *AMI MVScmdMgt
```

to

```
TOWER = SA *AON *MSM *Graphics *AMI MVScmdMgt
```

Editing the customization member AOFCUST

AOFCUST is the customization member for the sysplex-related recovery actions of msys for Operations. It:

- Tells msys for Operations which recovery actions are to be performed automatically.
- Supplies information that msys for Operations needs for some of these recovery actions.

Note: By default, all recovery actions are disabled.

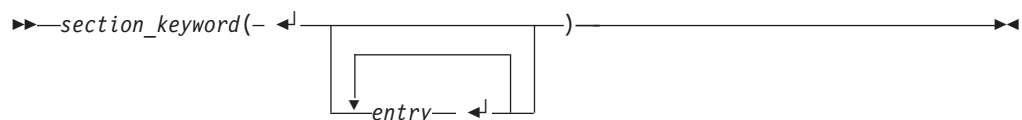
A sample of the AOFCUST member is contained in the ING.SINGNPRM library. It is recommended to edit a copy of this sample and copy it to the data set &Q1..DSIPARM specified in the startup procedure MSOAPROC. This data set must be shared in the sysplex so that all instances of msys for Operations within the sysplex are customized identically.

AOFCUST consists of several sections. The format of these sections is described by syntax diagrams. Note the following convention for these diagrams:

Line break symbol in the syntax diagrams

In the syntax diagrams of this chapter, the symbol ◀^l denotes a line break.

The format of a section is:



Every entry must be placed in a separate line. The closing parenthesis of a section must also be on a separate line. The maximum length of any line is 72 characters.

The entries can consist of only one keyword, but they can also have an internal structure of their own. You can comment out an entry by entering an asterisk in column 1.

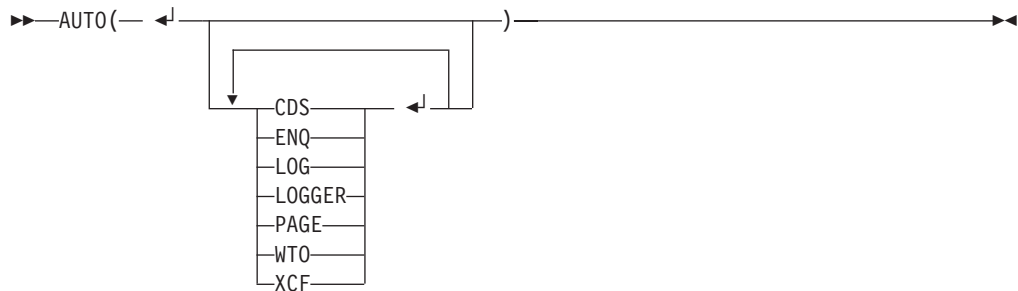
For checking the syntactical correctness of AOFCUST, msys for Operations provides the INGCUST command. For details, see “INGCUST” on page 155.

AUTO section – switching functions On and Off

Purpose

You can enable or disable the recovery functions of msys for Operations in the AUTO section. By default, all recovery actions are disabled.

Format



Parameters

CDS

Switches on background recovery of alternate CDSs. Automatic recreation of CDSs with INGPLEX CDS is always enabled. It cannot be switched off in the AUTO section. For general information, see “Ensuring continuous availability of alternate couple data sets” on page 9.

For recovery of alternate CDSs, you must specify the spare volumes on which msys for Operations may allocate the new alternate CDSs in the CDS section.

ENQ

Enables the handling of long-running ENQs.

LOG

Switches on recovery of the system log. For general information, see “Resolving a system log failure” on page 12.

LOGGER

Switches on the recovery actions relating to the system logger. For general information, see “Managing the system logger” on page 10.

Note that if you switch on automation for system logger recovery, you also activate the recovery function for alternate CDSs, even if CDS automation (CDS keyword) is switched off. Therefore, if you decide to use system logger automation, but do not want msys for Operations to recreate alternate CDSs for some or all CDS types, you must delete any spare volumes defined for the types in question.

PAGE

Prevents auxiliary storage shortages by predefining local page data sets.

WTO

Switches on recovery of WTO(R) buffer shortages. For general information, see “Resolving WTO(R) buffer shortages” on page 12.

For WTO(R) buffer recovery, specify the jobs that may be canceled in order to resolve the buffer shortage in the WTOBUF section.

XCF

Switches the automation of the IXC102A message on. If XCF is enabled make sure that the following conditions are met:

- msys for Operations must run on all systems in the sysplex because the message is issued only on one system
- The system must be enabled for this automation
- The BCP internal interface must be available
- The system being removed from the sysplex must be running on a G5 or follow on CPC

Because the automation must know where the system is located, to send the hardware command to the appropriate Support Element you must predefine your hardware configuration in the HW section of AOFCUST.

Furthermore if want to define other hardware commands than SYSRESET or to disable the automation for some systems you need to supply definitions in IXC102A section of AOFCUST.

Example

In the following example, the first system is re-IPLed when the automation takes place. The second system is deactivated, and the third system is not automated at all. All systems not specified are automated depending on the flag XCF in the AUTO section. For these systems the default action SYSRESET CLEAR is being performed.

```
IXC102A(  
  CMD(sys1,LOAD CLEAR)  
  CMD(sys2,DEACTIVATE)  
  DISABLE(sys3)  
)
```

COMMON section – common definitions

Purpose

This section defines values that are common to all functions of msys for Operations. Note that the TEMPHLQ parameter is required by the automation.

Format

```
►►COMMON(— ◀◀—TEMPHLQ—(—hlq—) — ◀◀—  
└──────────STCJOBNM—(—jobname—)────────┘ ◀◀—)◀◀
```

Parameters

TEMPHLQ

This keyword introduces the high-level qualifier which is used to assemble a data set name for allocating temporary data sets needed by programs running as started tasks.

hlq The high-level qualifier can consist of 17 characters. The name must comply to the MVS naming rules.

STCJOBNM

This keyword introduces the job name being used for programs running as started tasks. When not defined the job name of each started task defaults to the procedure name.

jobname

The job name may consist of up to 8 characters. The name must comply to the MVS job naming rules.

Example

```
COMMON(  
  TEMPHLQ (AOC.TEMP)  
  STCJOBNM(AOCSTC)  
)
```

PAGE section – predefined and dynamically allocated local page data sets

Purpose

The recovery prevents outages caused by auxiliary storage shortages.

To enable the function add the PAGE keyword to the existing section AUTO in the member AOFUCST. Note that also the PAGTOTL parameter defined in one of IEASYSxx PARMLIB members used during the IPL must specify a value greater than the number of local page data sets currently used. And, you must have defined at least one predefined local page data set or a spare volume including the high level qualifier.

Format

►►PAGE(— ◀ | Predefined Data Set | — ◀ | —) — ◀◀
 | Dynamically Allocated Data Set | | Job |

Predefined Data Set

|—DSN—(—*dsname*—)— ◀ |—————|

Dynamically Allocated Data Set

|—HLQ—(—*hlq*—)— ◀ |—CYL—(—*nnn*—)— ◀ | Volume List |—————|

Volume List

|—VOL—(—*volume*—)— ◀ |—————|

Job

|—JOB—(—^{*}—————, — CANCEL —) — ◀ |—————|
 | *jobname* [*] | | KEEP |

Parameters

CANCEL

This keyword indicates that the specified job is cancelled when the job caused the shortage and the shortage cannot be relieved within a specific time frame.

CYL

This keyword defines the space being used on dynamic allocations of page data sets.

DSN

This keyword defines the data set name of a predefined spare local page data set.

HLQ

This keyword defines the high level qualifier of dynamically allocated page data sets.

JOB

This keyword defines the job being checked when shortage condition occurs.

KEEP

This keyword indicates that the specified job must not be cancelled when the job caused the shortage.

VOL

This keyword defines a list of volumes where a local page data set can be allocated dynamically.

dsn

Is the name of a predefined local page data set. The data set must be allocated on a volume shared by all systems in the sysplex and catalogued in the master catalog. In case the systems in the sysplex do not share the master catalog the page data set must be recatalogued on each system except the one which created the data set using the following IDCAMS statement:

```
DEF PAGESPACE(NAME(dsn) VOLUME(volume) RECATALOG)
```

hlq

Is the high-level qualifier which is used to assemble a data set name for creating and allocating a page data set. The qualifier may consist of up to 23(!) characters and must comply to the MVS data set naming rules.

Note: The high level qualifier must point to the master catalog and must not be SMS managed.

jobname

Is the name of a job or a name pattern. Specify KEEP or CANCEL for jobs that can or cannot be cancelled. You can use wildcards when specifying the job name.

nmn

Is the space in cylinders (100 — 999) being used for the dynamic allocation of a local page data set.

volume

Is the serial number of a volume where a local page data set can be allocated.

Note: The volumes must be shared all systems in the sysplex.

Usage

To customize the automation, you can define the following:

- The predefined local page data sets that can be immediately added via PAGEADD.

- The potential volumes where new local page data sets can be allocated.
- The high-level qualifiers of those local page data sets being dynamically allocated. The actual data set name is built by concatenating the qualifier by '.V' followed by the volume serial number being used in the allocation.
- The amount of space for each dynamic allocation of a local page data set. If omitted a default value of 400 is used.

Note: On a 3390 DASD 100 cylinders are adequate to 70 MB. The formatting process lasts approximately 18 seconds for this amount of space.

- The names of jobs that can or cannot be cancelled. If no "JOB(*,...)" statement is defined, the following statement is generated:

```
JOB(*,KEEP)
```

All keywords except HLQ and CYL can be specified as often as needed. Up to 8 volume names can be specified per VOL statement. When both JOB(*,CANCEL) and JOB(*,KEEP) are specified the KEEP statement takes precedence.

The qualifiers of the HLQ statement may contain system symbols like &SYSNAME. If a symbol cannot be substituted it is replaced by its name. The qualifier is appended by the system name followed by '.Vvvvvvvv.Snn' to assemble the data set name where 'v' is the volume serial number and 'n' is a sequence number from 00 to 99.

The recommendation is to use predefined spare local page data sets rather than dynamically allocated data sets. This minimizes the time of the storage shortage. Each predefined local page data set should be allocated with 10% space of local page space currently used by the system.

Note: After a predefined spare local page data set has been made available by the automation to any system in the sysplex and it is no longer being used, you must delete the data set manually using the PAGEDEL to return the data set to pool of spare local page data sets.

Example

```
PAGE(
  DSN (SPARE.LOCAL.PAGE.DS1)
  DSN (SPARE.LOCAL.PAGE.DS2)
  HLQ (SPARE.DYNLOCAL.PAGE)
  CYL (600)
  VOL (vol111,vol1222,vol1333)
  VOL (vol1999)
  JOB (BTCH*,CANCEL)
  JOB (*,KEEP)
)
```

ENQ section – resources to be monitored and jobs to be cancelled

Purpose

This section defines the resources to be monitored when you want to automate long running ENQs. You must define the following:

- The resource(s) being monitored
- The time frame for each resource when an ENQ on the resource is treated as a long running ENQ

To prevent any outage you may define what jobs locking the resource can be cancelled when a long running ENQ is detected. You also can define if those jobs are being dumped before cancelled. You can define:

- The names of the jobs which should be cancelled or kept when detecting a long running ENQ
- The title of the dump taken before a job is cancelled
- The default storage areas to be dumped
- The symbol definitions to be used when the dump specifications are provided by a PARMLIB member

Format

```

>>ENQ(— Resource —
      DUMP(—sdata—) Job Symbol Definition TITLE(—title—)
>>—)
  
```

Resource

```

RES(—qname—,—rname—,—waittime—)
  
```

Job

```

JOB(—*—,—DUMP—)
     jobname[*]
     asid
     KEEP
     NODUMP
     xx[,xx]
  
```

Symbol Definition

```

SYMDEF(—*—,—[&]&symbol.—='value'.—)
        jobname[*]
        asid
  
```

Parameters

DUMP

This keyword defines the dump options being used when a dump is taken before the job is cancelled.

On the JOB statement it indicates that the specified job is cancelled with a dump using the default dump options when the job is the owner of a long running ENQ.

JOB

This keyword defines the job being checked when a long running ENQ is detected.

KEEP

This keyword indicates that the specified job must not be cancelled when the job is the owner of a long running ENQ.

NODUMP

This keyword indicates that the specified job is cancelled without a dump when the job is the owner of a long running ENQ.

RES

This keyword defines the resource to be monitored.

SYMDEF

This keyword defines the symbol being substituted when a dump is taken.

TITLE

This keyword defines the default dump title when a dump is taken.

asid

is the four-character address space ID, for example, 000A.

jobname

is the name of a job or a name pattern. You can use wildcards when specifying the job name.

qname

is the major resource name being checked. You can use wildcards when specifying the major resource name.

rname

is the minor resource name being checked. You can use wildcards when specifying the minor resource name.

sdata

specifies the storage areas to be dumped when the keyword DUMP is used instead of a PARMLIB member. If omitted the following areas are assumed:

CSA	Common Service Area
GRSQ	Global resource serialization (ENQ/DEQ/RESERVE) queues
RGN	Private area of address space being dumped, including LSQA (Local System Queue Area) and SWA (Scheduler Work Area)
SQA	System queue area
NOSUM	No summary dump
TRT	GTF, system trace, master trace, and NIP hardcopy buffer data

For further details of dump options refer to the description of the MVS DUMP command.

symbol

is the name of a system symbol to be substituted.

title

is the title of the dump. The title can be up to 100 characters in length. If omitted the title defaults to "Dump by msys for Operations due to a long ENQ detection".

value

is the substitution value.

waittime

is the time in seconds (30 — 999) after the automation treats an ENQ as a long blocking ENQ.

xx is the suffix of a IEADMCxx PARMLIB member describing the dump

specifications. This suffix is used on the dump command instead of the default dump options when a job is cancelled due a long running ENQ detection.

Usage

The syntax rules for the PARMLIB suffix(es) and the symbol definitions comply to the rules of the MVS DUMP command. For details see *MVS System Commands*.

The keywords RES, JOB, and SYMDEF can be specified as needed. Specifying KEEP on the JOB statement means this job must not be cancelled regardless of the locks being held. The remaining specifications DUMP, NODUMP, and the suffix(es) allow the cancellation of the job after the wait time has expired. NODUMP suppresses the dump before the job is cancelled.

Note that for using the ENQ automation function GRS must be active.

Example

```
COMMON(
  DUMP (CSA,RGN,SUM,GRSQ)
  JOB  (0001,KEEP)
  JOB  (CICS*,KEEP)
  JOB  (ABC*,NODUMP)
  JOB  (BTCH*,D0,D1)
  JOB  (*,DUMP)
  RES  (MAJOR1,MINOR1,30)
  RES  (MAJOR2,*,999)
  TITLE (Dump by automation due to a long ENQ detection)
)
```

CDS section – spare volumes for CDS recovery

Purpose

The volumes on which msys for Operations may allocate a new alternate CDS are specified in the CDS section. Every CDS type has its own pool of spare volumes. If you define no spare volumes for a CDS type, no recovery will be performed for this type even if CDS is set in the AUTO section.

The CDS section contains two types of entries, the high level qualifier for the new alternate CDSs, and one list of spare volumes for every CDS type. The list can contain up to eight volumes.

Format

►► CDS(— HLQ—hlq— Spare volumes)

Spare volumes

—VOL—(—SYSPLEX—,—volume_list—) —

CFRM
ARM
LOGR
SFM

Parameters

HLQ

This value keyword introduces the high-level qualifier for the data sets.

hlq The high-level qualifier can consist of one to three data set qualifiers. The name must comply to the MVS naming rules and can have up to 26 characters.

VOL

Every VOL entry specifies the spare volumes for one CDS type. Only one VOL entry can be specified per CDS type.

CDS_type

The CDS type must be the first element of the VOL entry.

volume_list

For every VOL entry, up to eight volumes can be specified. Note that the existence of these volumes is not checked by msys for Operations.

Usage

- The spare volumes must not be managed by SMS.
- In order to improve performance and availability you should associate spare volumes with CDS types according to the recommendations given in *z/OS MVS Setting Up a Sysplex*. It is recommended that you define three spare volumes for every CDS type for which alternate CDSs are to be (re)created automatically.

Example

```
CDS(
  HLQ AOC.CDS.TEST
  VOL (SYSPLEX,AOCLIB,AOCUSR,AOCBCK)
  VOL (CFRM,AOCUSR,AOCLIB,AOCBCK)
  VOL (LOGR,AOCLIB,AOCUSR,AOCBCK)
  VOL (SFM,AOCLIB,AOCUSR,AOCBCK)
  VOL (ARM,AOCLIB,AOCUSR,AOCBCK)
  VOL (WLM,AOCLIB,AOCUSR,AOCBCK)
)
```

HW section – hardware configuration

Purpose

Currently two automation functions need to know where a system or coupling facility is located to send the hardware command to the appropriate Support Element. Because there is no link between the system name and the LPAR name you must predefine your hardware configuration.

Format

►►HW(—◄◄| CPC | Image |)◄◄

CPC

↓
CPC(—cpcsyn—,—netid.nau—,—authtkn—)◄◄

Image

↓
IMAGE(—sysname—,—lparname—,—cpcsyn—,—plexname—,—CF—)
MVS
OTHER◄◄

Parameters

CPC

This keyword defines the Central Processor Complex (CPC).

cpcsyn

is the synonym character string for the Central Processor Complex (CPC). The maximum length is eight characters.

netid.nau

is the network address of the Support Element. Obtain this information from the CPC Support Element or the HMCs where the CPC is defined.

authkn

is the authorization information for communications. The CPC support element, BCP internal interface configuration provides this information. The authorization value is the BCP internal interface community name. msys for Operations can process only *authkn* specifications in uppercase letters. Therefore, it may be necessary to change your Support Element BCP internal interface community name field.

IMAGE

defines a single system and its linkage to a CPC.

sysname

Is the name of the software image (system). For MVS systems it is the MVS system name defined. For a coupling facility (CF) it is the coupling facility name defined. For other systems use the name that identifies them in your system configuration.

lparname

is the name of a Logical PARTition (hardware image), the system defined to run on the CPC specified with *cpcsyn*.

Note: When defining a CPC running in basic mode define the LPAR name must as identical to the CPC name.

plexname

is the name of the sysplex where the MVS system *sysname* is defined as a member. If *sysname* identifies a coupling facility, refer to the coupling definitions to identify the sysplexname the coupling facility is related to. If *sysname* is not a member of an MVS sysplex, omit the parameter but leave the comma.

MVS|CF|OTHER

Choose MVS if *sysname* has a z/OS or OS/390 operating system. Choose CF if *sysname* is a coupling facility, use OTHER for any other operating system.

Example

```
HW(
CPC  (FREEWAY,DEIBMD1.X7F1E30A,PUBLIC)
CPC  (SAFOS ,DEIBMD1.X7F1F20A,PUBLIC)
CPC  (YORAMA ,DEIBMD1.X7E1FA0A,PUBLIC)
IMAGE (CF1 ,CF1 ,FREEWAY ,SYSPLEX ,CF )
IMAGE (CF2 ,CFF ,FREEWAY ,SYSPLEX ,CF )
IMAGE (CFD ,CFD ,FREEWAY ,CIM7PLEX,CF )
IMAGE (CFx ,CFE ,FREEWAY ,SYSPLEX ,CF )
IMAGE (CIM7 ,CIM7 ,FREEWAY ,CIM7PLEX,MVS )
```


Note:

The following restriction applies to the processor operations commands ACTIVATE and LOAD:

Both commands invoke processor functions, which can cause asynchronous events such as operator messages at BCP (Basic Control Program) internal interface initialization time or processor hardware wait states. Currently, the BCP internal interface does not allow to monitor and control these events.

The command is being sent to the Support Element before the outstanding WTOR is replied.

Note: The commands above are disruptive commands. Because the Support Element still considers the affected image as operating it rejects any disruptive commands if not otherwise stated. This requires the option FORCE being specified which is automatically appended by the automation routine.

ENABLE

This keyword defines the system(s) that are enabled for the automation of the message IXC102A.

DISABLE

This keyword defines the system(s) that are disabled for the automation of the message IXC102A.

Example

In the following example, the first system is re-IPLed when the automation takes place. The second system is deactivated, and the third system is not automated at all. All systems not specified are automated depending on the flag XCF in the AUTO section. For these systems the default action SYSRESET CLEAR is being performed.

```
IXC102A(
  CMD(sys1,LOAD CLEAR)
  CMD(sys2,DEACTIVATE)
  DISABLE(sys3)
)
```

WTOBUF section – jobs to be cancelled in case of buffer shortage

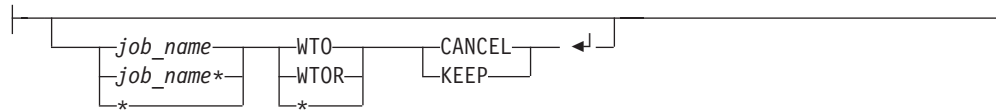
Purpose

In the WTOBUF section, you specify which jobs can be cancelled in case of a WTO or WTOR buffer shortage. Every entry of this section states that a job or group of jobs must be kept or cancelled when a WTO and/or WTOR buffer shortage occurs.

Format

```
➤➤ WTOBUF ( — ◀ | Jobs | — ) ➤➤
```

Jobs



Parameters

job_name

Specifies a job or group of jobs to be kept or cancelled. You specify a group of jobs by using an asterisk (*) as a placeholder for all jobs or for jobs whose name starts with a given string (for example, 'abc*' for all job names starting with 'abc').

buffer_type

Possible values are:

WTO	WTO buffer
WTOR	WTOR buffer
*	Both buffer types

processing_option

Possible values are

CANCEL

The job or group of jobs is to be cancelled for the specified buffer type.

KEEP

The job or group of jobs is to be kept for the specified buffer type.

Usage

msys for Operations reads the entries of this section downward and applies the first fitting entry it finds, regardless of whether more specific fitting entries occur farther down in the list. Thus, when the first entry is * * CANCEL, and the second is ABCD * KEEP, then job ABCD will be cancelled. All jobs for which no fitting entry is found will be kept.

Example

In the following example, all jobs whose name starts with 'ABC' are kept, with the only exception of the job ABCD. All other jobs are cancelled.

```
WTOBUF(  
  ABCD * CANCEL  
  ABC* * KEEP  
  * * CANCEL  
)
```

Chapter 4. Making security definitions

This chapter describes how to protect your system resources from unauthorized access and how to implement a security concept that grants operators access to those commands and resources they need. The recommended method to do this in msys for Operations is to use a Security Access Facility (SAF) product, such as Resource Access Control Facility (RACF).

An alternative method to protect your resources and commands without using an SAF product is described in “Defining security using msys for Operations definitions” on page 191.

Activating security classes

A CNMSAF1 sample file is delivered with msys for Operations in the CNMSAMP library. The sample predefines security settings for your use which enable you to make a quick start. To make use of the sample, copy the CNMSAF1 file to your DSIPARM library. The following assumes that you either tailor the file to your enterprise’s requirements, or that you issue the necessary commands from the system console.

To restrict which operators can log on to the msys for Operations domain, to restrict which commands the operators are authorized to issue, and to ensure that the msys for Operations domain starts properly, the following SAF classes must be activated:

- APPL
- NETCMDS
- NETSPAN

To activate these classes, issue the following commands:

```
SETOPTS CLASSACT(APPL)
SETOPTS CLASSACT(NETCMDS) GRPLIST
SETOPTS CLASSACT(NETSPAN)
```

To protect the msys for Operations domain from unauthorized access, issue the following command to define the msys for Operations domain name to previously defined RACF class APPL:

```
RDEFINE APPL domain_name UACC(NONE)
```

where *domain_name* is the domain name specified in the msys for Operations startup procedure MSOAPROC (INGNVAP0).

Defining operators, passwords, and logon attributes

This section provides the following information about operator security:

- Operator identifiers and passwords
- Operator logon attributes

Defining operator identifiers and passwords

Define a unique operator identifier for each operator who logs on to msys for Operations by changing or adding an ADDUSER statement in sample CNMSAF1.

Defining operators, passwords, and logon attributes

For example, change:

```
ADDUSER OPER1 PASSWORD(user1)
```

to

```
ADDUSER NEWOPER PASSWORD(PW1)
```

In this example, NEWOPER is the operator identifier and PW1 is the initial password. The first time NEWOPER logs on to msys for Operations, the password must be changed.

Note that the names of msys for Operations commands, components, printers (hardcopy logs), terminals, or task identifiers should not be used for operator identifiers. Also, do not use the following reserved keywords:

ALL	NNT
DPR	OPT
DST	OST
HCL	PPT
HCT	SYSOP
LOG	TCT
MNT	

An operator can change their password from the msys for Operations logon panel when an SAF product is being used for password authorization.

Defining tasks to RACF

The PPT and SSIR tasks must be defined to RACF.

For the PPT task, change the domain name in CNMSAF1 in the following command to match the domain name that is specified in your msys for Operations startup procedure. For example, if the domain name is MSO01 change:

```
ADDUSER domain_namePPT
```

to

```
ADDUSER MSO01PPT
```

For the SSIR task, change the task name you defined (as described in “Defining the name of the automation router task” on page 34) to match the task name in the following command. For example, if the domain name is MSO01 change:

```
ADDUSER domainSIR
```

to

```
ADDUSER MSO01SIR
```

Use the ADDUSER and ALTUSER commands for all of the msys for Operations autotasks exactly as specified in sample CNMSAF1.

Defining operator logon attributes in the NVSS segment of an SAF product

Operator logon attributes describe logon attributes that are associated with an operator. Operator logon attributes are defined in the NVSS segment of an SAF product. The following operator attributes are supported:

Defining operators, passwords, and logon attributes

CTL	CTL defines an operator's authority to control resources. CTL(GLOBAL) is recommended to enable operators to control all msys for Operations resources.
IC	IC specifies the initial command list to run when an operator logs on. Initial command list LOGPROF1 defines PF key definitions and a unique MVS console name. It is recommended to use LOGPROF1 for all operators except automated operators (autotasks).
MSGRECV	MSGRECV specifies whether an operator is eligible to receive unsolicited messages that are not routed to a particular operator using either the ASSIGN command or msys for Operations automation. Possible values for MSGRECV are either YES or NO. Usually, the first operator to be logged on with MSGRECV(YES) is the authorized receiver of unsolicited messages.

In RACF, operator logon attributes are defined using the ALTUSER command. For example:

```
ALTUSER NEWUSER NETVIEW(IC(LOGPROF1) CTL(GLOBAL) MSGRECV(YES))
```

When the operator NEWUSER logs on:

- The initial command list LOGPROF1 is run
- Operator NEWUSER controls all msys for Operations resources
- Operator NEWUSER can receive unsolicited messages

Use the RACF PERMIT command to enable operators to log on to the domain name specified in your msys for Operations startup procedure. For example, to enable operator NEWOPER to log on to domain MSO01, specify the following RACF command in CNMSAF1:

```
PERMIT MSO01 CLASS(APPL) ID(NEWOPER) ACCESS(READ)
```

where MSO01 is the domain name and NEWOPER is the operator ID. Instead of an operator ID you can also specify a group, for example MSYSOPS1.

Command authorization

This section provides the following information about command authorization:

- Overview
- Command authorization using an SAF product such as RACF
- Command authorization for specific commands
- Additional recommendations about command authorization

Overview

Command authorization is the process of protecting commands from unauthorized use and selectively grant access to them.

Use RACF, or a comparable SAF product, to restrict access to commands, keywords, and values, and to grant operator access to them. You do this by defining the commands, keywords, and values as resources in the NETCMDS class of the SAF product, and then selectively granting operator access. This is most easily done by grouping operators into groups that correspond to their roles and level of expertise. The next step is to connect your users to those groups.

Command authorization

You can also specify commands, keywords, and values that are accessible universally. When you make these changes, you can have them take effect by requesting the SAF product to refresh the NETCMDS class definitions. You do not have to issue msys for Operations commands to include the changes.

In the CNMSAF1 sample the following groups are predefined:

MSYSOPS0

Users listed in this group are allowed to execute administrative commands.

MSYSOPS1

Users listed in this group are allowed to execute FORCE and REBUILD actions on structures.

MSYSOPS2

Users listed in this group are allowed to execute the SETXCF command with parameter ACOUPLE and PSWITCH

MSYSOPS3

Users listed in this group are allowed to execute the full functionality of INGCF ENABLE and INGCF DRAIN

MSYSOPS4

Users listed in this group are allowed to execute most restricted base NVSS commands

MSYSOPS5

Users listed in this group are allowed to execute all commands

Command authorization using an SAF product

The following sections describe the steps that are necessary for authorizing commands using an SAF product such as RACF.

Defining msys for Operations commands as NETCMDS resources

To define msys for Operations commands as resources in the NETCMDS commands class, use resource names as described in the following.

Commands are checked separately from keywords and values. When defining resource names, remember that the command is checked first. Commands, keywords, and value combinations are checked in the following order:

netid.luname.command
netid.luname.command.keyword
netid.luname.command.keyword.value

Where:

netid

Indicates the VTAM network identifier. You can specify a generic character (*) for this field.

This value is compared with the VTAM network identifier from the last activation of VTAM or +NONE+ if VTAM has not been activated. If you do not need to differentiate between *netids* and are not concerned about whether VTAM has been active, specify a generic character (*) for this field.

luname

Indicates the domain name for an instance of an msys for Operations program.

command

Indicates the command name on the CMDMDL statement in the DSICMSYS

member of DSIPARM, or a command list name. This must be the actual command name and not a synonym defined by the CMDSYN statement. No checking is done to validate that *command* is a valid command or command list name.

keyword

Indicates the keyword identifier which is protected.

value

Indicates the value identifier which is protected when used with the keyword on the command.

The keyword or value used with the command may not match the keyword or value being protected because of synonyms, defaults, and substitutions of values in the resource name.

Examples of NETCMDS resource definitions: The following examples show how to define NETCMDS resources using the RDEFINE command of RACF. RDEFINE can be abbreviated to RDEF.

- Create one resource in the NETCMDS class for each command you want to protect. To protect the CF keyword of the INGCF command, specify the following in the CNMSAF1 sample:

```
RDEF NETCMDS *.*.INGRCCHK.INGCF.CF UACC(NONE)
```

INGRCCHK is an internal routine that is called by the INGCF and INGPlex commands and needs to be specified.

- To minimize the number of occasions where SAF cannot achieve a command authorization decision, you can universally grant or deny access to the remaining commands, keywords, and values by defining a generic resource name for msys for Operations. Using RACF, you can do this for a *netid* of NETA and an *luname* of MSO01 by issuing one of the following commands:

```
RDEFINE NETCMDS *.*.* UACC(READ)
RDEFINE NETCMDS *.*.* UACC(NONE)
```

To allow an msys for Operations operator to issue an msys for Operations command protected in the NETCMDS class, you must grant a level of access of at least READ.

Summary—how to perform command authorization using SAF

The following step-by-step procedure shows an example (based on the CNMSAF1 sample file) of how to define operator authority to RACF, assuming the operators are already defined to RACF.

1. To activate the NETCMDS class, if not already active, specify the following:
SETROPTS CLASSACT(NETCMDS) GRPLIST
2. To define the NETCMDS class as a GENERIC class to allow the use of generic characters, if generic characters will be used, specify the following:
SETROPTS GENERIC(NETCMDS)
3. To define groups of operators, specify the following:
ADDGROUP MSYSOPS1
CONNECT user1 GROUP(MSYSOPS1) UACC(READ)
CONNECT user2 GROUP(MSYSOPS1) UACC(READ)
ADDGROUP MSYSOPS2
CONNECT user3 GROUP(MSYSOPS2) UACC(READ)
CONNECT user4 GROUP(MSYSOPS2) UACC(READ)
ADDGROUP MSYSOPS3
CONNECT user5 GROUP(MSYSOPS3) UACC(READ)

Command authorization

```
CONNECT user6 GROUP(MSYSOPS3) UACC(READ)
ADDGROUP MSYSOPS4
CONNECT user7 GROUP(MSYSOPS4) UACC(READ)
CONNECT user8 GROUP(MSYSOPS4) UACC(READ)
ADDGROUP MSYSOPS5
CONNECT user9 GROUP(MSYSOPS5) UACC(READ)
CONNECT user10 GROUP(MSYSOPS5) UACC(READ)
```

4. Define the commands, keywords and values to be protected.

To define the software-related functions of INGPLEX and INGCF as resources in the NETCMDS class, specify the following:

```
RDEFINE NETCMDS *.*.INGRRCHK.INGCF.STR UACC(NONE)
RDEFINE NETCMDS *.*.INGRRCHK.INGPLEX.CDS UACC(NONE)
RDEFINE NETCMDS *.*.INGRRCHK.INGCF.CF UACC(NONE)
RDEFINE NETCMDS *.*.INGRRCHK.INGPLEX.HW UACC(NONE)
```

5. To associate operator groups with command resources, specify the following:

```
PERMIT *.*.INGRRCHK.INGCF.STR CLASS(NETCMDS) ID(MSYSOPS1) ACC(READ)
PERMIT *.*.INGRRCHK.INGPLEX.CDS CLASS(NETCMDS) ID(MSYSOPS2) ACC(READ)
PERMIT *.*.INGRRCHK.INGPLEX.CDS CLASS(NETCMDS) ID(MSYSOPS3) ACC(READ)
PERMIT *.*.INGRRCHK.INGCF.CF CLASS(NETCMDS) ID(MSYSOPS3) ACC(READ)
PERMIT *.*.INGRRCHK.INGPLEX.HW CLASS(NETCMDS) ID(MSYSOPS4) ACC(READ)
```

In this example:

- User1 and user2 can only rebuild or force a selected structure with INGCF STRUCTURE
- User3 and user4 are allowed to rebuild or force a selected structure, to switch policies and to manipulate couple data sets
- User5 and user6 can rebuild and force a selected structure, set the sender path of a coupling facility ONLINE or OFFLINE, switch policies, manipulate couple data sets.

Protecting immediate commands when CMDAUTH=SAF

Immediate commands are host msys for Operations commands which are defined with TYPE=I on the CMDMDL statement in DSICMSYS, or defined with TYPE=B on the CMDMDL statement in DSICMSYS and have been entered from the command line. These commands are run under the control of an IRB exit. This environment prohibits the use of an SAF RACROUTE macro to call a security product.

For msys for Operations, the only immediate command that must be protected is the CLOSE IMMED command. To define which operator can stop msys for Operations by issuing the CLOSE IMMED command, add the operator identifier to the sample backup table, CNMSBAK1, provided with msys for Operations in the CNMSAMP library. Copy CNMSBAK1 to your DSIPARM library. For example, to protect the CLOSE command keep the statements predefined in sample member CNMSBAK1:

```
GROUP MSYSOPS5 NETOP1, NETOP2
PROTECT *.*.CLOSE
```

To authorize the MSYSOPS5 group to use the CLOSE command keep the statement in the CNMSBAK1 sample:

```
PERMIT MSYSOPS5 *.*.CLOSE
```

Additional recommendations for command authorization

Recommended commands to protect

Customers must decide which commands to protect based on their specific security requirements. However, it is recommended that you restrict the following commands, because they can affect the msys for Operations environment or access to it:

- AFTER (Use of the PPT keyword.)
- AT (Use of the PPT keyword.)
- AUTOTBL
- CHRON (Use of the ROUTE keyword. See “Defining security for the CHRON command” on page 58 for more information.)
- CLOSE
- DEFAULTS
- EVERY (Use of the PPT keyword)
- EXCMD. (See “Defining EXCMD command authorization” on page 62 for more information.)
- FOCALPT
- GETCONID
- GLOBALV
- INGRCHK
- MODIFY
- MVS (See “Defining additional MVS command authority” on page 62 for more information.)
- OVERRIDE
- PURGE
- READSEC (See “NVSS READSEC and WRITESEC commands” on page 66 for more information.)
- REFRESH
- RMTCMD
- RUNCMD
- SUBMIT (See “Defining SUBMIT command authorization” on page 63 for more information.)
- SETCONID
- START
- STOP
- VARY
- WRITESEC (See “NVSS READSEC and WRITESEC commands” on page 66 for more information.)

Exceptions to command authorization checking

Major exceptions to command authorization checking include:

- Commands entered as replies to the msys for Operations WTOR (message DSI802A) are not authority checked. To prevent users from issuing commands using the WTOR, specify CMDWTOR=NO in the MVSPARM statement in DSIDMN. This prevents msys for Operations from issuing the WTOR.
- Command authority checks are not made against the PPT or DST tasks. Therefore, you need not authorize these tasks to access your protected commands.
- Commands issued from a source ID of *BYPASS* are not checked for command authorization by:
 - The SAF product OPERCMDS class
 - The SAF product NETCMDS class

Command authorization

The SOURCEID will default to *BYPASS* if the command was entered at an extended multiple console support (EMCS) console and the operator was not logged on to the EMCS console.

Auditing command authority checking

You can audit access to protected commands, keywords, and values. This auditing can be done on an individual command, keyword, or value basis.

You can audit access to SAF-defined resources. You can control this auditing on a resource basis. For each resource, you can specify whether to perform no auditing, to audit authorization failures, to audit authorization successes, or to audit all access attempts whether successful or not. For RACF, the auditing level is specified using the RDEFINE or RALTER commands when you define the resource name. Additionally, to allow msys for Operations commands in general to be audited, you must ensure that the RACF SETROPTS statement specifies AUDIT(NETCMDS). RACF generates SMF records that contain details at the audit level you specify for commands. You can then use the RACF report writer to create reports that describe attempts to access RACF-protected resources. For more information on the RACF report writer, refer to the RACF library.

The more auditing you request the SAF product to perform, the more system resources are required by the SAF product. You need to determine the value of the audit level you choose versus the expense in system overhead, both processor and DASD.

Protecting commands containing special characters

There are some special characters that cannot be included in the command identifier or SAF resource name. For this reason, msys for Operations translates these special characters to other characters before passing them to either the msys for Operations command authorization table or the SAF product. The special characters that are translated along with their translated results are:

Reserved Character	Translated Result
.	/
*	+
%	?
&	:
- (dash)	_ (underscore)
' ' (blank)	_ (underscore)

For example, the following msys for Operations command can be entered by an msys for Operations operator:

```
LIST MEMSTAT=.*
```

To restrict access to this keyword and value using RACF, include the following RACF profile:

```
LIST MEMSTAT=.*
```

Note that the asterisk was translated to a plus.

Determining the user identity used for authority checking commands

Authority checking restricts the ability of an operator or a task to use commands, keywords, and values.

Table 4 identifies the operator or task identifier that is used to authority check msys for Operations commands based on the command and environment. The identity is referred to as the SOURCEID.

Table 4. SOURCEID determination

Command and environment	SOURCEID determination
EXCMD command or a same-domain LABEL command prefix used to queue an imbedded command to another task.	The SOURCEID is the task that issued the EXCMD command, or the existing SOURCEID at the time the EXCMD command was issued.
TIMER commands that are scheduled to run under the PPT.	The SOURCEID is the task that issued the AT, EVERY, CHRON, or AFTER command, or the existing SOURCEID at the time the AT, EVERY, CHRON, or AFTER command was issued. Note: The SOURCEID is not destroyed by saving and restoring timer commands.
msys for Operations SUBMIT command for jobs submitted to the operating system from msys for Operations.	If OPERSEC=SAFDEF or OPERSEC=SAFCHECK, the identity that is checked by the operating system is the issuer of the SUBMIT command, or the existing SOURCEID at the time the SUBMIT command was issued. For other values of OPERSEC, msys for Operations' authority is used for submitting the job.
msys for Operations commands that were entered at an MVS operator console.	When an MVS console has been associated with an autotask using the AUTOTASK command with the CONSOLE= parameter, msys for Operations commands can be entered from that MVS console. This is done by prefixing the msys for Operations command with the msys for Operations designator character, which by default is %. If the MVS operator has logged on to the MVS console with a user ID, the SOURCEID is the user ID of the MVS operator. If an operator has not logged on at the EMCS console, the SOURCEID of that task defaults to *BYPASS*. Commands issued from a source ID of *BYPASS* are not checked for command authorization by: <ul style="list-style-type: none"> • The msys for Operations command authorization table • The SAF product OPERCMDS class • The SAF product NETCMDS class Note: If a command is entered from the MVS master console, it will be routed to one of the following: <ul style="list-style-type: none"> • The autotask with the specific console name • The autotask with console name "*MASTER*" • The autotask with console name "*ANY*"

Command authorization

Table 4. *SOURCEID* determination (continued)

Command and environment	SOURCEID determination
msys for Operations commands that are entered using the MVS MODIFY command.	<p>When an MVS console has been associated with an autotask using the AUTOTASK command with the CONSOLE= parameter, msys for Operations commands can be entered from that MVS console by issuing an MVS MODIFY or STOP command against the msys for Operations task. The msys for Operations command is entered as text following the MODIFY command. The first parameter on the MODIFY command is the application ID that is being modified. If the MVS operator has logged on to the MVS console with a user ID, the SOURCEID is the user ID of the MVS operator.</p> <p>If an operator has not logged on at the EMCS console, the SOURCEID of that task defaults to *BYPASS*. Commands issued from a source ID of *BYPASS* are not checked for command authorization by:</p> <ul style="list-style-type: none">• The msys for Operations command authorization table• The SAF product OPERCMDS class• The SAF product NETCMDS class <p>Note: If a command is entered from the MVS master console, it will be routed to:</p> <ul style="list-style-type: none">• The autotask with the specific console name• The autotask with console name <i>"*MASTER*"</i>• The autotask with console name <i>"*ANY*"</i>
msys for Operations commands that were entered by TSO users.	<p>When a TSO user ID has been associated with an autotask using the AUTOTASK command with the CONSOLE= parameter, msys for Operations commands can be entered from that TSO user ID when the user is acting as an MVS operator by using an EMCS console session, or when using SDSF. The SOURCEID is the TSO user's user ID.</p>
Commands issued from JCL.	<p>When a job that issues a msys for Operations command is submitted by a TSO user ID, the SOURCEID is the TSO user ID. If the ID of the submitter is unknown, a default user ID is inserted. The value of the default user ID is defined by the system installation.</p>
MVS ROUTE command issued from msys for Operations.	<p>If the MVS command ROUTE is issued from a msys for Operations task, the originating source ID is always passed to the SAF product for authorization checks in the OPERCMDS class. This occurs for all settings of AUTHCHK and CMDAUTH.</p>
Commands that are routed to an operator from the automation table.	<p>The SOURCEID is the operator ID to which the command is routed.</p> <p>Note: Commands from the automation table are subject to authority checking unless SEC=BY was specified on the CMDMDL statement or SEC=DE was specified (or SEC was not specified) and AUTOSEC=BYPASS is in effect. For more information, refer to the DEFAULTS command in the msys for Operations online help.</p>

Understanding security for specific commands

This section provides additional information about protecting the following commands:

- CHRON
- EXCMD
- MVS
- SUBMIT

Defining security for the CHRON command

The CHRON command has a syntax that is more complex than most commands. CHRON uses multiple levels of keywords, items in lists, and quoted strings.

Command security for the CHRON command is checked so that operands within parentheses can be uniquely defined in an SAF product.

The following rules describe CHRON commands and which command identifiers are checked:

RULE 1: Each keyword that does not take a value (NOSAVE, SAVE, LOCAL, GMT, REFRESH, TEST, and DEBUG) is checked in the form:

Command example:

```
netid.luname.CHRON.keyword
```

RULE 2: Each keyword with a value is checked in the form:

```
netid.luname.CHRON.keyword.value
```

With the CHRON command, the value may be a list or quoted string.

Command example:

```
CHRON AT=(),RECOVERY=IGNORE,NOSAVE,LOCAL,ROUTE=OPER1,ID=TEST1,COMMAND='MSG ALL HELLO'
```

The following command identifiers are checked:

```
netid.luname.CHRON
netid.luname.CHRON.AT.()
netid.luname.CHRON.RECOVERY.IGNORE
netid.luname.CHRON.NOSAVE
netid.luname.CHRON.LOCAL
netid.luname.CHRON.ROUTE.OPER1
netid.luname.CHRON.ID.TEST1
netid.luname.CHRON.COMMAND.'MSG_ALL_HELLO'
```

Rule 3A: Keywords appearing within parenthesized lists of other keywords are checked using the hierarchy of keywords with a "(" between so that the keyword hierarchy can be uniquely identified. The compound keyword that is generated is tested with the value of the innermost keyword. This checking is done at each level of the nesting of the lists. When a keyword is within a list that is the value of another keyword, the notation uses both keywords with a "(" between them.

Rule 3B: From the outermost to innermost, if a "keyword=(list)" appears, if any values appear in the list without keywords, the "keyword=value" check is done for that value. The keyword that is checked is the keyword hierarchy defined by Rule 3A.

Command example:

```
CHRON EVERY=(INTERVAL=(000-01.00.00 FOR=08.00.00))
```

The following command identifiers are checked:

```
netid.luname.CHRON
netid.luname.CHRON.EVERY.(INTERVAL=(000_01/00/00_FOR=08/00/00))
netid.luname.CHRON.EVERY(INTERVAL.(000_01/00/00_FOR=08/00/00)
netid.luname.CHRON.EVERY(INTERVAL.000_01/00/00
netid.luname.CHRON.EVERY(INTERVAL(FOR.08/00/00
```

Substitution of certain special characters is performed as described in "Protecting commands containing special characters" on page 56. For example, a dash becomes an underscore in the command identifier.

Command authorization

Rule 4: Quoted string values are checked as a single value, including the apostrophes and all text within the apostrophes.

Command example:

```
netid.luname.CHRON.REM.'ISN'T THIS A REMARK STRING?'
```

The following command identifier is checked:

```
CHRON REM='ISN'T THIS A REMARK STRING?'
```

Rule 5: For the DAYSWEET keyword, days of the week can be followed by a sublist identifying particular weeks of the month. The day name and each item in the sublist are treated as a unit.

Command example:

```
CHRON EVERY=(DAYSWEET=(NOT MON(1ST 2nd)))
```

The following command identifiers are checked:

```
netid.luname.CHRON
netid.luname.CHRON.EVERY.(DAYSWEET=(NOT MON(1ST 2ND)))
netid.luname.CHRON.EVERY(DAYSWEET.(NOT MON(1ST 2ND)))
netid.luname.CHRON.EVERY(DAYSWEET.NOT
netid.luname.EVERY(DAYSWEET.MON(1ST)
netid.luname.EVERY(DAYSWEET.MON(2ND))
```

This lets you check the sublist values without concern for the order of the items within the sublist. Notice that the value "MON(1st 2nd)" is not checked since the values MON(1st) and MON(2nd) are checked.

The following table illustrates a detailed list of possible command identifiers that may be defined for the CHRON command. The rule that causes the command identifier to be checked is shown in the second column.

Table 5. Command identifiers for the CHRON command

Commands and keywords identifier	RULE	SAF resource identifier
CHRON	Command Name	netid.luname.CHRON
AT=	2 2 3B 3B 2	netid.luname.CHRON.AT() netid.luname.CHRON.AT.(timespec datespec) ¹ netid.luname.CHRON.AT.timespec netid.luname.CHRON.AT.datespec ¹ netid.luname.CHRON.AT.yyy_mm_dd_hh/mm/ss/micros ¹
AFTER=	2 2	netid.luname.CHRON.AFTER.timespec ¹ netid.luname.CHRON.AFTER.ddd_hh/mm/ss/micros ¹
EVERY=	2 2 2	netid.luname.CHRON.EVERY.NONE netid.luname.CHRON.EVERY.() netid.luname.CHRON.EVERY.(everyoptions) ¹

Table 5. Command identifiers for the CHRON command (continued)

Commands and keywords identifier	RULE	SAF resource identifier
EVERY=(INTERVAL=	3A 3B 3B 3A	netid.luname.CHRON.EVERY(INTERVAL.() netid.luname.CHRON.EVERY(INTERVAL.(intervaloptions) ¹ netid.luname.CHRON.EVERY(INTERVAL.timespec ¹ netid.luname.CHRON.EVERY(INTERVAL. ddd_hh/mm/ss/micros ¹
EVERY=(INTERVAL=(FOR=	3A 3A	netid.luname.CHRON.EVERY(INTERVAL(FOR.timespec netid.luname.CHRON.EVERY(INTERVAL(FOR. hh/mm/ss/micros ¹
EVERY=(INTERVAL=(MXREPEAT=	3A 3A	netid.luname.CHRON.EVERY(INTERVAL(MXREPEAT. NOLIMIT netid.luname.CHRON.EVERY(INTERVAL(MXREPEAT. repeat_count
EVERY=(INTERVAL=(OFF=	3A 3A	netid.luname.CHRON.EVERY(INTERVAL(OFF.timespec netid.luname.CHRON.EVERY(INTERVAL(OFF. hh/mm/ss/micros ¹
EVERY=(REMOVE=	3A 3A, 3B 3B 3B 3A	netid.luname.CHRON.EVERY(REMOVE.MANUALLY netid.luname.CHRON.EVERY(REMOVE.(removeoptions) ¹ netid.luname.CHRON.EVERY(REMOVE.datespec ¹ netid.luname.CHRON.EVERY(REMOVE.timespec ¹ netid.luname.CHRON.EVERY(REMOVE. yyyy_mm_dd_hh/mm/ss/micros ¹
EVERY= (REMAFTER=	3A 3A	netid.luname.CHRON.EVERY(REMAFTER.timespec ¹ netid.luname.CHRON.EVERY(REMAFTER. ddd_hh/mm/ss/micros ¹
EVERY= (DAYSWEEK=	3A 3B 3B 3B 5	netid.luname.CHRON.EVERY(DAYSWEEK.ALL netid.luname.CHRON.EVERY(DAYSWEEK.(daysweeklist) ¹ netid.luname.CHRON.EVERY(DAYSWEEK.NOT netid.luname.CHRON.EVERY(DAYSWEEK.dayname netid.luname.CHRON.EVERY(DAYSWEEK. dayname (sublist_element) ¹
EVERY=(DAYSMON=	3A 3B 3B 3B	netid.luname.CHRON.EVERY(DAYSMON.ALL netid.luname.CHRON.EVERY(DAYSMON.(dayslist) ¹ netid.luname.CHRON.EVERY(DAYSMON.NOT netid.luname.CHRON.EVERY(DAYSMON.dayofmonth ¹
EVERY=(CALENDAR=	3A 3B 3B 3B	netid.luname.CHRON.EVERY(CALENDAR.ALL netid.luname.CHRON.EVERY(CALENDAR.(calendarlist) ¹ netid.luname.CHRON.EVERY(CALENDAR.NOT netid.luname.CHRON.EVERY(CALENDAR.keyname ¹
RECOVERY=	2 2 2	netid.luname.CHRON.RECOVERY.IGNORE netid.luname.CHRON.RECOVERY.AUTOLGN netid.luname.CHRON.RECOVERY.PURGE
SAVE	1	netid.luname.CHRON.SAVE

Command authorization

Table 5. Command identifiers for the CHRON command (continued)

Commands and keywords identifier	RULE	SAF resource identifier
NOSAVE	1	netid.luname.CHRON.NOSAVE
LOCAL	1	netid.luname.CHRON.LOCAL
ID=	2	netid.luname.CHRON.ID.idname
NOTIFY=	2	netid.luname.CHRON.NOTIFY.(notifylists)
NOTIFY=(PURGE=	3B	netid.luname.CHRON.NOTIFY(PURGE.(purgelist)
	3B	netid.luname.CHRON.NOTIFY(PURGE.taskname
NOTIFY=(REMOVE=	3B	netid.luname.CHRON.NOTIFY(REMOVE.(removelist)
	3B	netid.luname.CHRON.NOTIFY(REMOVE.taskname
NOTIFY=(IGNORE=	3B	netid.luname.CHRON.NOTIFY(IGNORE.(ignorelist)
	3B	netid.luname.CHRON.NOTIFY(IGNORE.taskname
NOTIFY=(RUN=	3B	netid.luname.CHRON.NOTIFY(RUN.(runlist)
	3B	netid.luname.CHRON.NOTIFY(RUN.taskname
REFRESH	1	netid.luname.CHRON.REFRESH
TEST	1	netid.luname.CHRON.TEST
DEBUG	1	netid.luname.CHRON.DEBUG
COMMAND=	4	netid.luname.CHRON.COMMAND.'quoted string' ¹
REM=	4	netid.luname.REM.'quoted string' ¹

Defining EXCMD command authorization

The msys for Operations EXCMD command is used to send commands to another task.

There are two operands that are used when issuing the EXCMD command. One is the *operator_id* where the command is being sent, and the other is the *command* being sent. These two operands are checked as a keyword-value pair.

Note: When protecting the target verb of EXCMD, specify the command verb, not any synonym. Unless otherwise documented, the verb is the label used on the CMDMDL statement. The verb for labeled commands beginning with a slash is EXCMD.

For example, the command identifier to protect EXCMD OPER1 LOGOFF is:

```
PROTECT *.*.EXCMD.OPER1.LOGOFF
```

For information about the EXCMD command refer to the online help.

Defining additional MVS command authority

You can protect individual MVS system commands from unauthorized use with the OPERCMDS class of an SAF product, such as RACF. This is additional

1. This value may have a special character, such as "." or "-", for example in the programmer time notation. You substitute the character "/" for "." and "_" for "-" when making the security definition.

authorization checking done at the MVS level, after the command security checking done by the NETCMDS class of an SAF product.

To protect MVS commands:

1. Ensure your OPERSEC setting has a value of SAFCHECK or SAFDEF.
2. Define command profiles to restrict specific commands from operators. For example, to restrict all operators from being able to issue an MVS QUIESCE command, enter:

```
RDEFINE OPERCMDS MVS.QUIESCE UACC(NONE)
```
3. Ensure that the OPERCMDS class is active and enabled for processing. The following RACF commands can be used to do this:

```
SETROPTS CLASSACT(OPERCMDS)  
SETROPTS RACLIST(OPERCMDS)
```
4. When the OPERCMDS class is active, use the RACF REFRESH function when you change a definition:

```
SETROPTS RACLIST(OPERCMDS) REFRESH
```

Defining SUBMIT command authorization

You can protect jobs submitted from NVSS using the SUBMIT command. When the NVSS SUBMIT command is issued, you have three layers of protection that you can use:

1. The SUBMIT command can be protected using NVSS command authorization. This is your first layer of protection. By protecting at this level, you can stop the processing for unauthorized users before the job is ever submitted to the system.
2. For jobs that reside in data sets that are NOT part of the DSIPARM data set concatenation, you can use the SAF DATASET class to prevent users from accessing those data sets. Using the SAF DATASET class prevents users from submitting jobs that are members of those data sets. This is the second layer of protection. An attempt to access the data set is made before the job is actually submitted.
3. The SAF JESJOBS class can be used to prevent users from submitting specific jobs. This is effective for DSIPARM and non-DSIPARM data sets. This is the third layer of protection. This check happens after the job has been submitted to JES, (not synchronously with the msys for Operations SUBMIT command).

Note: A failure at this level will not be reported back to the msys for Operations console. The JESJOBS class failure is only reported to the master console/syslog.

Controlling access to data sets and members

To prevent unauthorized changes of data, you can protect data sets with an SAF product, such as RACF. To prevent unauthorized viewing of passwords and other restricted information, protect them with msys for Operations commands such as READSEC and WRITESEC. See “NVSS READSEC and WRITESEC commands” on page 66 for recommendations.

Data set security

You can restrict unauthorized alteration of data sets from the msys for Operations environment using the DATASET class of the security product. The following are some considerations when using the DATASET class of the security product:

Controlling access to data sets and members

- msys for Operations requires CONTROL access to the DSILOG data set to write to the netlog.
- msys for Operations requires READ access to the first data set identified by the DSILIST DD statement.
- msys for Operations requires READ access to non-DSIPARM data sets that are specified with the msys for Operations SUBMIT command.
- Each of the following NVSS commands require UPDATE access to the first data set identified by the DSILIST DD statement.
 - AUTOTBL (with the LISTING keyword)
 - AUTOCNT (with the FILE keyword)
 - QRYGLOBL (with the FILE keyword)

Note: NVSS trace records are not made for calls to the DATASET class, because the calls are made by MVS for the NVSS tasks.

Restricting and granting access to data sets

To activate the data set protection described in the preceding section, do the following:

1. Add profiles for the data sets you want to protect. The RACF product requires that the highest-level qualifier of the data set name be either a task or group name.

For example, use the RACF ADDSD command to add data set profiles. From an authorized TSO user, enter the following command to protect the OPER1.STATS data set:

```
ADDSD 'OPER1.STATS'
```

2. Authorize the operator tasks so they can access the data set. For example, use the RACF PERMIT command to authorize operator tasks to the data set. To authorize NETOP1 to have update access to OPER1.STATS, enter the following command from an authorized TSO user ID:

```
PERMIT 'OPER1.STATS' CLASS(DATASET) ID(NETOP1) ACCESS(UPDATE)
```

Granting NVSS and the STC-user access to data sets

Access to XCF utilities: The CDS recovery as well as some operator commands use the XCF utilities to retrieve couple data set information. Because the DD name SYSPRINT is required by the utilities, but can also be assigned by NVSS for holding log data, the call of the utilities is implemented as a started task in the PROCLIB. The input and output data sets used by the started tasks are dynamically allocated and deleted by the NVSS address space. This requires the RACF ALTER access to these data sets for NVSS (*nvuserid*).

When the address space of the started task is created, the operating system assigns a user ID (IBM default: *stcuser*) to the started task. This user ID must have RACF UPDATE access to the data sets. To add a data set profile specify the following:

```
ADDSD 'hlq.*.HSA*.*' UACC(NONE)
```

where:

hlq is the high-level qualifier of the automation status file used by the current NVSS

HSA* is the NVSS domain

To grant the *stcuser* and the *nvuserid* access to the data set, specify, for example:

```
PE 'hlq.*.*.*' CLASS(DATASET) ACC(UPDATE) ID(stcuser)
PE 'hlq.*.*.*' CLASS(DATASET) ACC(ALTER) ID(nvuserid)
```

Controlling access to data sets and members

Access to HOM Interface: Sometimes after an IPL an operating system does not know its sender paths to the coupling facilities in the sysplex. In this case the automation functions call the HCD HOM interface to determine the missing path information. As the HOM interface must not run authorized the interface is called via a started task. The input and output data sets used by the started tasks are dynamically allocated and deleted by the NVSS address space. This requires the RACF ALTER access to these data sets for NVSS.

When the address space of the started task is created, the operating system assigns a user ID (IBM default: STCUSER) to the started task. This user ID must have RACF UPDATE access to the data sets. The data set names are created as follows:

```
hlq.domain.HSAyyddd.Xhhmmss
```

where:

hlq	is the high-level qualifier for temporary data set defined during the customization
domain	is the domain ID of the current NVSS
X	O or P

For an example of how to do your RACF definitions refer to “Access to XCF utilities” on page 64.

Access to IPL Information: The new automation function collecting, displaying, comparing, and deleting IPL information uses two started tasks. The first started task runs immediately after an IPL as part of COMMNDxx list processing, and collects the IPL information in the msys for Operations VSAM data set “IPLDATA”. The remaining functions are handled by a NVSS command. Since the started task as well as the command can delete IPL information both need RACF CONTROL access to the VSAM data set. The started task collecting the information needs RACF read access to all parmlib members.

When a comparison of IPL information is requested the NVSS command schedules the second started task to call ISRSUPC — the compare utility provided by ISPF — as this utility requires fixed ddname. The input and output data sets used by the second started tasks are dynamically allocated and deleted by the NVSS address space. This requires the RACF ALTER access to these data sets for NVSS.

When the address space of the started task is created, the operating system assigns a user ID (IBM default: STCUSER) to the started task. This user ID must have RACF UPDATE access to the data sets. The data set names are created as follows:

```
hlq.domain.opid.INGPIPLx
```

where:

hlq	is the high-level qualifier for temporary data set defined during the customization
domain	is the domain ID of the current NVSS
opid	is the NVSS operator ID
x	L, N, or O

For an example of how to do your RACF definitions refer to “Access to XCF utilities” on page 64.

Controlling access to data sets and members

Access to CDS spare volumes: Because the CDS recovery allocates and deletes spare couple data sets via an XCF utility the user ID assigned to the started task address space must also have RACF ALTER access to these couple data sets. The data set names are created as follows:

hlq123.cdstype.CDSnn

where:

hlq123 is the high-level qualifier defined during customization

cdstype is ARM, CFRM, LOGR, SFM, SYSPLEX

To create a data set profile and to grant the *stcuser* access specify, for example:

```
ADDSD 'hlq.**.*' UACC(NONE)
PE 'hlq.**.*' CLASS(DATASET) ACC(ALTER) ID(stcuser)
```

Access to user-defined couple data sets: In addition, the user ID of the started task address space needs RACF READ access to all user-defined couple data sets. To add a user-defined CDS profile and grant the *stcuser* access specify, for example:

```
ADDSD 'sys1.cds.**' UACC(NONE)
PE 'sys1.cds.**' CLASS(DATASET) ACC(READ) ID(stcuser)
```

When LOGR recovery is enabled, the user ID needs RACF ALTER access to the LOGR couple data sets as well. For example, specify the following:

```
ADDSD 'sys1.cds.log.**' UACC(NONE)
PE 'sys1.cds.log.**' CLASS(DATASET) ACC(ALTER) ID(stcuser)
```

Access to Spare Local Page Data Sets: The new auxiliary shortage recovery allocates and formats spare page data sets. For this reason NetView requires RACF ALTER access to these page data sets. The names of the spare page data set are built as follows:

hlq.sysname.Vvolume.Snn

where:

hlq is the high-level qualifier for page data sets defined during the customization

sysname is the name of system for which the data set is allocated

volume is the serial number of the volume on which the data set is allocated

nn is a unique sequence number

For an example of how to do your RACF definitions refer to “Access to XCF utilities” on page 64.

NVSS READSEC and WRITESEC commands

Use the msys for Operations READSEC and WRITESEC commands to restrict access to data sets and members by msys for Operations commands. When you specify security for the READSEC command, it affects all of the msys for Operations commands which can display sensitive information, such as:

- BROWSE with a member name
- NCCF LIST with the CLIST or PROFILE keywords
- PIPE stages
 - < (From disk)
 - QSAM

Controlling access to data sets and members

- VSAM command DSIVSMX

Using READSEC and WRITESEC is the only way to prevent operators from viewing data sets and members using these msys for Operations commands. In msys for Operations, security is defined so that operators have access to DSIOPEN and msys for Operations online help. DSIOPEN is a DD name designed to hold information which should not be secured, such as NEWS data and PF key definitions. Anything other than DSIOPEN and online help may be considered sensitive information.

Because attempts to define security for these commands is considered a severe error, message BNH115A is generated every time an operator logs on. The error text for this message is "SPECIAL SECURITY IN EFFECT FOR BROWSE AND READSEC", which indicates msys for Operations has defined default protection for sensitive data sets and members, and the msys for Operations commands which display data sets or members will fail. You must delete any security definitions for the commands and reinitialize msys for Operations to clear the error condition.

If you use command authorization without specifying values for READSEC and WRITESEC, operators will have access to all data sets and members.

Do not protect DD name CNMPNL1, operators need to access online help that is contained there.

For more information about how to use the READSEC and WRITESEC commands, refer to the online help.

Controlling Access to the Processor Hardware Functions

The following describes what must be defined in a SAF product such as RACF to enable the usage of the SNMP over the BCP internal interface, in the following referred as HSAET32. Note, that this interface is needed if you have enabled CF and XCF automation in your AOFCUST definitions.

Enabling NVSS

Before using msys for Operations for CF or XCF automation, NVSS must be authorized for HSAET32.

1. Define resource HSA.ET32OAN.HSAET32 in the CLASS FACILITY
2. Permit NVSS READ ACCESS to this facility class resource. The following example shows the RACF commands used to define the resource and to grant READ access to it for NVSS.

```
SETROPTS CLASSACT(FACILITY)
RDEFINE FACILITY HSA.ET32OAN.HSAET32 UACC(NONE)
PERMIT HSA.ET32OAN.HSAET32 CLASS(FACILITY) ID(stcuser) ACC(READ)
```

With the SETROPTS command, the RACF class FACILITY is made available. The second command RDEFINE fully qualifies the HSAET32 resource and sets an universal access of "NONE". With the PERMIT command, the RACF defined user stcuser gets READ access to this resource. Userid stcuser must be the user ID associated with your NVSS started task. Note, that with RACF you may specify the resource more generically if that is suitable for your environment.

Access to the CPCs

Each CPC in your AOFCUST definition must have a corresponding resource profile defined with your SAF product. The skeleton of the CPC resource is:

Controlling access to data sets and members

HSA.ET32TGT.netid.nau

The netid.nau part of the resource name corresponds with the netid.nau definition of the CPC entry in your AOFCUST definition. The period between netid and nau is part of the resource name. The following example shows how to define a CPC resource in RACF.

```
RDEFINE FACILITY HSA.ET32TGT.DEIBMD1.X7F1F30A UACC(NONE)
```

The CPC with netid DEIBMD1 and nau X7F1F30A is defined as a resource in the RACF class Facility with a universal access attribute of NONE. Note, that with RACF you may specify the resource more generically if that is suitable for your environment.

Levels of CPC Access

The following lists the access levels and their meaning for the CPC resources.

READ

Retrieves, gets configuration information from the CPC

WRITE

Updates, sets configuration information of the CPC

CONTROL

Performs Operations Management Commands of the CPC

Note that this access level scheme is for the CPC and its LPAR levels.

Defining the CPC Access Level

With msys for Operations, individual NVSS users and NVSS autotasks need to be authorized to access the CPCs that are defined in your AOFCUST file. The current HSAET32 interface allows the user to perform the HW functions indirectly. No direct command interface is provided allowing the user to issue security level dependant commands. For this reason, the highest possible access level CONTROL has to be given to all NVSS users entitled to perform CF enable and CF drain functions for a specific CPC. In addition, the following NVSS autotasks need to be authorized with access level CONTROL for all defined CPCs:

- The XCF autotasks
- The autotasks defined with SYN %AOFOPXCFOPER% in automation table member AOFMSGAO

The AUTXCFxx autotasks and the additional ones from %AOFOPXCFOPER% are used internally if INGCF DRAIN or INGCF ENABLE is invoked by an authorized user. The IXC102A message automation is also performed by these autotasks. The following example shows how to permit access to a CPC resource in RACF.

```
PERMIT HSA.ET32TGT.DEIBMD1.X7F1F30A CLASS(FACILITY) ID(AUTXCF) ACC(CONTROL)
```

The XCF autotask AUTXCF gets access level CONTROL for the CPC resource DEIBMD1.X7F1F30A.

LPAR access example:

```
PERMIT HSA.ET32TGT.DEIBMD1.X7F1F30A.* CLASS(FACILITY) ID(AUTXCF) ACC(CONTROL)
```

The XCF autotask AUTXCF gets access level CONTROL for the CPC resource DEIBMD1.X7F1F30A and all its defined logical partitions.

Chapter 5. Activating msys for Operations

If you have completed all of the steps in the previous chapters, you are ready to start the msys for Operations program.

Starting msys for Operations Using the MSOAPROC Startup Procedure

Enter the following at the system console:

```
S MSOAPROC
```

You see messages similar to those in Figure 3.

```
Display Filter View Print Options Help
-----
SDSF SYSLOG 6110.101 KEY2 KEY2 06/28/2001 2W 10,016 COLUMNS 51 130
COMMAND INPUT ==> SCROLL ==> CSR
0080 AOF532I 06:35:56 : AUTOMATION ENVIRONMENT HAS BEEN INITIALIZED 134
0080 AOF869I 0 ERROR(S) ENCOUNTERED PROCESSING MEMBER ACFZ999 135
0290 D C
0090 IEE889I 06.35.56 CONSOLE DISPLAY 137
0090 MSG: CURR=3 LIM=9999 RPLY:CURR=3 LIM=99 SYS=KEY2 PFK=02
0090 CONSOLE/ALT ID ----- SPECIFICATIONS -----
0090 SYSLOG COND=H AUTH=CMDS NBUF=0 UD=N
0090 ROUTCDE=ALL
0090 NO CONSOLES MEET SPECIFIED CRITERIA
0290 D SMF
0090 IEE974I 06.35.56 SMF DATA SETS 139
0090 NAME VOLSER SIZE(BLKS) %FULL STATUS
0090 P-SYS1.KEY2.MAN1 KEY2PP 9000 88 ACTIVE
0090 S-SYS1.KEY2.MAN2 KEY2PP 9000 0 ALTERNATE
0090 S-SYS1.KEY2.MAN3 KEY2PP 9000 0 ALTERNATE
0080 AOF511I 06:35:56 : ACFZ999 AUTOMATION CONTROL FILE COMMON VALUES 140
0080 HAVE BEEN INITIALIZED
0080 AOF540I 06:35:57 : INITIALIZATION RELATED PROCESSING HAS BEEN 141
0080 COMPLETED.
0290 IEA630I OPERATOR AAUTO2Y2 NOW ACTIVE, SYSTEM=KEY2, LU=AUTO2
DSI802A IPSNN REPLY WITH VALID NCCF SYSTEM OPERATOR COMMAND
F1=HELP F2=SPLIT F3=END F4=RETURN F5=IFIND F6=BOOK
F7=UP F8=DOWN F9=SWAP F10=LEFT F11=RIGHT F12=RETRIEVE
```

Figure 3. Message for Starting msys for Operations

The important messages are DSI802A, AOF532I, and AOF540I.

The WTOR message DSI802A looks as follows:

```
*0003 DSI802A CNM01 REPLY WITH VALID NCCF SYSTEM OPERATOR COMMAND
```

This message indicates that NVSS has been started properly. You need not reply to this message as long as msys for Operations is supposed to run.

When you want to shutdown msys for Operations reply CLOSE IMMED to DSI802A. Note that you must be authorized to issue that reply.

The other two messages look similar to the following:

```
AOF532I hh:mm:ss AUTOMATION ENVIRONMENT HAS BEEN INITIALIZED
AOF540I hh:mm:ss INITIALIZATION RELATED PROCESSING HAS BEEN COMPLETED
```

These messages indicate that initialization of the Sysplex Functions part has been successful.

Logging On to msys for Operations

To log on to msys for Operations, enter:

```
LOGON APPLID(domain_name) LOGMODE(logmode)
```

A logon panel is displayed, where you must specify one of the operator IDs defined before (see “Defining operators, passwords, and logon attributes” on page 49), and the appropriate initial password.

When your input has been accepted, you must change the password. Then the main menu of msys for Operations is displayed. For details concerning the logon procedure, see Chapter 7, “Logging on to msys for Operations” on page 75.

Chapter 6. Configuring msys for Operations for Your Environment

The following sections explain how to configure certain functions of msys for Operations for your environment.

Defining Passwords for VSAM Databases

You can define security passwords for the VSAM databases that are used by certain NVSS tasks. The general procedure for this is as follows:

1. Stop the task.
2. Modify the definition statements in INGALLC0 that define the database, changing them to include the specification of VSAM cluster passwords. Rerun job INGALLC0 using these modified statements to delete and redefine the database.
3. Update the initialization member in DSIPARM that is associated with the task by specifying the password for the password parameters.
4. Restart the task.

The following table lists the VSAM clusters for which this is possible:

Table 6.

VSAM cluster	Task	DSIPARM member	Parameter
NETVIEW.VnRnMn.xxxxx.DSILOGP	DSILOG	DSILOGBK	PPASS
NETVIEW.VnRnMn.xxxxx.DSILOGS	DSILOG	DSILOGBK	SPASS
NETVIEW.VnRnMn.xxxxx.DSISVRT	DSISVRT	DSISVRTD	PPASS

Printing the Network Log and Trace Log

The member CNMSJM04 in NETVIEW.VnRnMn.V1R4M0.CNMSAMP contains a sample print job for these logs. Copy this member to SYS1.PROCLIB and rename it to CNMPRT. If you defined passwords for the network log and the trace log, add a password statement to job CNMPRT.

To change the defaults used to print the network or trace logs, control statements must be passed to PGM=DSIPRT using the DSIINP DD statement. You can do this using one of two methods:

1. Create the following statements for a job stream or an instream procedure:

```
//DSIINP DD *  
        PASSWD=password  
        OPER1,OPER2,NETOP1  
        TRANSTBL MOD=DSIEBCDC
```

2. Create a statement similar to the following to define a data set member to contain the print control statements and put the preceding print control statements in this member.

```
//DSIINP DD DSN=SYS1.PARMLIB(MEMBER),DISP=SHR
```

Only the second method applies for system-started JCL procedures.

Part 3. Using the operator interface of msys for Operations

This part contains the following topics:

- How to log on to msys for Operations
- The parts of the message display screen
- How to issue commands
- How to move between the msys for Operations components
- How to list your program function key definitions
- How to use the netlog
- How to get help

Chapter 7. Logging on to msys for Operations

1. To log on to msys for Operations, enter:

```
logon applid(applid) logmode(logmode)
```

where *applid* is the domain name of the msys for Operations application to which you are logging on. LOGMODE is an optional parameter that specifies information about your terminal session.

When you log on, msys for Operations queries the device for screen size and color attributes if the logmode specifies to issue the query. Otherwise, msys for Operations uses the screen size specified in the logmode. The command facility adapts to use the entire width and depth of the screen. All components of the msys for Operations program support color where the display is capable of displaying color.

When a session is established, a msys for Operations logon panel similar to the one shown in Figure 4 is displayed.

```

NN  NN          VV      VV
NNN NN  EEEEE  TTTTTT  VV      VV  II  EEEEE  WW      WW  TM
NNNN NN  EE      TT      VV      VV  II  EE      WW  W  WW
NN NN NN  EEEE  TT      VV      VV  II  EEEE  WW  WWW  WW
NN  NNNN  EE      TT      VV  VV  II  EE      WWW  WWW
NN  NNN  EEEEE  TT      VVV      II  EEEEE  WW  WW
NN  NN
                        V

5697-B82 (C) Copyright Tivoli Systems 1986, 2001 - All Rights Reserved
U.S. Government users restricted rights - Use, duplication, or disclosure
restricted by GSA ADP schedule contract with IBM corporation.
Licensed materials - Property of Tivoli Systems.
Domain = NTV74          Tivoli NetView V1R4

OPERATOR ID ==>          or LOGOFF
PASSWORD ==>
HARDCOPY LOG ==>         device name, or NO, default=NO
RUN INITIAL COMMAND ==>  YES or NO, default=YES
Takeover session ==>     YES or NO, default=NO

Enter logon information or PF3/PF15 to logoff
```

Figure 4. Example of msys for Operations logon panel

2. Type your operator identification (for example, OPER1) in the space next to the OPERATOR ID field, where the cursor is located.

Blanks entered in the msys for Operations logon fields will be treated as null characters. For example, OPER 1 entered in the OPERATOR ID field of the msys for Operations logon screen will be treated as OPER1 because the blank between "R" and "1" is treated as a null character.

3. Enter your password. If you want to change your password, leave this field blank.
4. If you do not want to use an initial command, type no in the RUN INITIAL COMMAND field. If you want to use an initial command, leave this field blank or type yes. The initial command is set up by your system programmer to eliminate some manual procedures.

5. Press Enter. A panel similar to Figure 5 is displayed.

```
NVSS V1R4 IPSN7 02/07      Tivoli NetView  IPSN7 SCHR      06/07/01 14:08:08 H
- IPSN7      DSI020I OPERATOR SCHR LOGGED ON FROM TERMINAL X7626A02 USING
  PROFILE (NVSSPRO5 ), HCL ( )
- IPSN7      LOGPNVSS
- IPSN7      DSI083I AUTOWRAP STOPPED
C IPSN7      CNM357I PFKDEF : PF KEY SETTINGS NOW ESTABLISHED. 'DISPFK' TO SEE
  YOUR PF KEY SETTINGS
! IPSN7

  Enter LOG or LOGOFF to terminate session.
  Enter HELP to obtain help.
  Lead operator has been notified of your logon.
  To obtain help from the NETWORK CONTROL CENTER, enter

      MSG PPT, your question here

! IPSN7
News for January 1, 2001

  The operating system now contains some NetView functions.  For more
  information on NetView, see http://www.tivoli.com/nv390
-----
??? *** DSI662I SCREEN HELD
```

Figure 5. *msys for Operations news panel*

6. Press the CLEAR or ENTER key to go to the msys for Operations Main Menu. Possibly, you must do this more than once. After msys for Operations processes the operator profile, the main menu is displayed:

```
CNM1NETV                      z/OS msys for Operations                      Main Menu

                          Operator ID = SCHR      Application = IPSN7020

Enter a command (shown highlighted or in white) and press Enter.

Browse Facility              BROWSE command
Command Facility            NCCF command
News                        NEWS command
PF Key Settings             DISPFK command
Help Facility               HELP command
Index of help topics        INDEX command

To log off or disconnect    LOGOFF command or DISC command

TO SEE YOUR KEY SETTINGS, ENTER 'DISPFK'
Action==>
```

Figure 6. *msys for Operations main menu*

7. The main menu contains a command line from which you can issue commands to msys for Operations.

Chapter 8. The message display screen

To access the message display screen, type `nccf` on the main menu panel and press ENTER. You can also use the ROLL function (PF6 by default). For further information about the ROLL function refer to “Moving between the Components” on page 83. In order to return to the main menu, enter `mainmenu` in the command entry area or use the ROLL function again.

The layout of the message display screen is as follows:

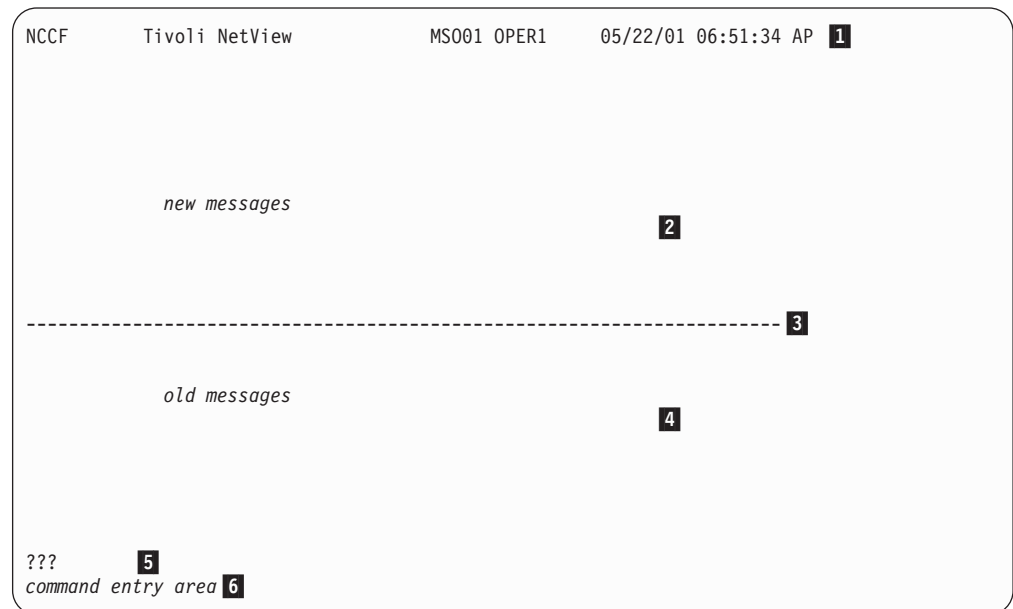


Figure 7. Sample message display screen

Session identification line

The first line of the panel, identified with **1**, gives you the name of the panel that appears. The next field lists the domain name (MS001) and your operator identifier (OPER1). The next two fields list the current date and time. The last two fields contain a combination of A, H, P, W, or a blank, which indicates whether messages can be written to the panel. The A, H, P, and W indicators are described in the following list:

- A** The autowrap indicator means that AUTOWRAP is active. If autowrap is on and the display is full of data, it is automatically overlaid with new data. If autowrap is not on, press the Clear or Enter key to allow new data to overlay the display screen.
- H** The held-screen indicator means that the screen does not roll forward unless it is unlocked by the operator. You can use this indicator if you need time to read the screen before it is erased, or to freeze the screen while you mark messages for deletion or enter a command.
- P** The pause status indicator. A command list running on the operator task is pausing for operator input, and will not continue until the operator enters information.

W The wait indicator. A command list running on the operator task is waiting for messages or other events, such as for a specified amount of time to elapse.

Message area

The message area displays commands, responses, and messages from the system. Figure 8 shows a sample display screen.

```
NCCF                                Tivoli NetView      MS001 OPER1    05/22/01 21:41:49
* MS001  OPER1  D NET,ID=NCP98
MS001  OPER1  IST097I  DISPLAY  ACCEPTED
' MS001  OPER1
IST075I  NAME = NCP98                , TYPE = PU T4/5
IST486I  STATUS= ACTIV              , DESIRED STATE= ACTIV
IST247I  LOAD/DUMP PROCEDURE STATUS = RESET
IST484I  SUBAREA = 98
IST391I  ADJ LINK STATION = 014-S    , LINE = 014-L    , NODE = NTC0VTAM
IST654I  I/O TRACE = OFF, BUFFER TRACE = OFF
IST077I  SIO = 00040374 CUA = 014
IST675I  VR = 0, TP = 2
IST314I  END

-----
IST080I  J0032055 ACTIV              J0032057 ACTIV              J0032059 ACTIV
IST080I  J003205B ACTIV              J003205D ACTIV              J003205F ACTIV
IST080I  J0032061 ACTIV              J0032063 ACTIV              J0032065 ACTIV
IST080I  J0032067 ACTIV              J0032069 ACTIV              J003206B ACTIV
IST080I  J003206D ACTIV              A19CA01  ACTIV----E A19CA02  ACTIV----E
IST080I  A19CA03  ACTIV----E A19CA04  ACTIV----E
IST314I  END
???
```

Figure 8. Sample display screen

The dashed line, indicated by **3** separates the latest messages from the older ones. The messages are continually updated. You can use this line to locate the most recent messages. The most recent message is the one directly above the line, in the area indicated by **2**. The older messages displayed on the screen are below the line, in the area indicated by **4**. The oldest message is the one directly below the line.

The first line of the sample screen indicates that the VTAM command D NET,ID=NCP98 was issued by OPER1 on domain MSO01. The second line indicates that VTAM has accepted the command. The rest of the upper part of the screen consists of the output of the DISPLAY command. The codes in the first column of the first three lines indicate the type of the respective message. For the meaning of the type codes, see Table 8 on page 86.

Generally, messages disappear as the screen scrolls. Examples of exceptions include reply messages, held messages, and windowed responses.

Reply messages

Reply messages are messages to which you should reply before you delete them from the display screen. These messages appear in high intensity on your display screen with a reply number. Unsolicited reply messages received on the system console remain outstanding even after a reply is given. Delete these messages manually using the MVS control (K) command.

Held messages

Held messages are messages that are defined to be held on the screen. These messages appear in high intensity (or are otherwise highlighted) and appear at the top of the message area. Specific action must be taken to remove them, such as:

- De-emphasizing them with a Delete Operator Messages (DOM) command
- Specifically deleting them (by the operator)

The DOM command causes messages to lose highlighting immediately. This means they can now scroll off the screen. If there are more messages being held than can be displayed on your type of terminal, message DSI151I appears and the messages are queued. The queued messages appear only when existing ones are deleted.

To delete one or more held messages:

1. Move the cursor to the message line, using either the cursor keys or the TAB key.
2. To delete a single message, press **Enter**. The cursor will return to the command entry area.
3. To delete multiple messages, erase the first line of each message to be deleted (you can use the Erase EOF key) and press **Enter**. The cursor will return to the command entry area.

Attention: If an autowrap timeout occurs while you are typing over message text, that text might be moved or refreshed, thus destroying the typing that has been done.

To avoid losing information from the command entry area, you can either:

- Turn autowrap off, using the AUTOWRAP NO command.
- Use the HOLD command.

Windowed responses

Windowed responses are messages that are displayed in a scrollable window using the msys for Operations WINDOW command. This prevents the message responses from being overwritten by subsequent messages, and enables you to navigate through the information using standard BROWSE commands. For a description of the behavior of windowed responses, refer to the WINDOW command in the msys for Operations online help.

Response area

Near the bottom of the screen is a line that begins with the ??? indicator. This line is the response area, indicated by **5** in Figure 7 on page 77. Look here for error messages.

The =X= indicator is displayed in place of the ??? indicator when messages are arriving (prior to entering or after leaving a panel). This indicator means that only a limited set of commands can be used. Some of the commands you can use are:

- AUTOWRAP
- LOGOFF

Command entry area

The cursor is located in the command entry area, indicated by **6** in Figure 7 on page 77. You communicate with the msys for Operations program by entering commands here or you can invoke another msys for Operations component (for example, by entering `mainmenu`).

Chapter 9. Issuing commands

You can issue commands from the message display screen or from the menu. If you press a key on a terminal that has no keyboard buffering capability, and the controller is already processing a request from the host, the key is rejected, and the keyboard can lock up. You can then press **RESET** to unlock the keyboard and enable input to proceed.

The length of the command entry area is limited to three lines of 80 characters each. For input modes of two or three lines, on screens wider than 80 characters, the msys for Operations program indicates the end of the input area with three less-than symbols (<<<). When you press any action key (ENTER, any PF or PA key, or CLEAR), the command area is erased.

Repeating commands

The RETRIEVE command tells the system to place the last command you entered on the command line. If necessary, you can alter the command on the command line, or leave it as it is, then press ENTER to send the command to the system.

You can repeat the RETRIEVE command several times to display the last few commands that you sent to the system. The easiest way to use the RETRIEVE command is by assigning it to a PF key. The msys for Operations-supplied default for the RETRIEVE command is PF12.

Issuing MVS system commands

To issue commands to MVS, use the MVS command, which enables you to control MVS system operations without using a separate screen for multiple console support (MCS).

To issue a command from the msys for Operations command facility, enter MVS followed by a valid MVS command. For example, to display a list of active MVS tasks, enter:

```
mv s d a,l
```

msys for Operations displays the response from MVS on the message display screen.

Chapter 10. Moving between the components and using function keys

This chapter describes how you can move between the components of msys for Operations, and how to use and display function keys.

Moving between the Components

The msys for Operations program allows you to have multiple components active at the same time. For example, the message display screen (which is always active), the main menu, and a output panel of the BROWSE command can be active simultaneously. One method to move from one component to another is to enter the component name. Thus, you can enter `nccf` on the command line of the BROWSE panel to move to the message display panel.

You can also use the ROLL function to move among active components in a continuous loop. The PF key for ROLL that is supplied by msys for Operations is PF6. If your PF key settings have PF6 set to ROLL, then pressing PF6 returns you to the last panel you viewed in an active component.

Using program function keys

You can use program function (PF) or program access (PA) keys to send commands to the system. Doing so can save time because you do not have to type a command and then press the Enter key.

Listing PF and PA keys

Most PF and PA keys have already been set for you, with unique settings by component. They are set to commands that you will use quite often.

To display the current settings for the message display screen PF and PA keys, enter:

```
dispfk nccf
```

You can also display PF key settings for other components, such as log browse, by specifying their component abbreviations on the DISPFK command or a PF key set to that command. For example, the msys for Operations defaults specify the DISPFK command with the APPEND keyword as PF4, allowing you to type a component name on the command line, then press PF4 to see that component's PF keys. Browse the CNMKEYS member or enter `dispfk all` to display all PF key settings.

You can change the key settings for your actual session with the SET command; see "SET" on page 101.

Chapter 11. Using the netlog

The netlog is the record of the terminal activity that has occurred on the system. You can send commands, responses, and messages to the netlog. Each message contains the time and date it was sent and the names of the operator and domain it came from.

The netlog is stored in two VSAM data sets. One of these is the primary (referred to as netlogp), and the other the secondary log data set (referred to as netlogs). Only one log data set is used at any point in time. This is the active log data set (referred to as netloga). The other one is inactive (referred to as netlogi). At first, the primary data set is active, and the secondary inactive. When the primary log data set becomes full, logging is continued on the secondary log data set. Now the secondary data set is active, and the primary is inactive. When the secondary data set becomes also full, logging switches back to the primary data set; the old entries will be overwritten.

You can print the inactive netlog file in batch mode, while the system is using the active file as the log.

Displaying the netlog

You can use the BROWSE command to display a particular netlog data set. You can select the active or inactive log, or you can name the specific log (primary or secondary) to browse. For example, to display the active log, enter:

```
browse netloga
```

If the primary log data set is the active one, the command

```
browse netlogp
```

has the same effect.

You can specify a time and date range to limit the amount of netlog information displayed. For example, to display the primary netlog from 1:00 p.m. on 4/07/01 to 8:30 a.m. on 4/08/01, enter:

```
browse netlogp from 4/07/01 13:00 to 4/08/01 8:30
```

Note: If you specify a time range for browsing the netlog, the first and the last record of the specified time range remains the first and the last record during the entire browse.

You can use the FIND or ALL commands to locate specific information while you are browsing the netlog. For example, to find the words INVALID COMMAND, enter:

```
f 'invalid command'
```

Structure of the log entries

The following figure shows a netlog panel with a short explanation of the meaning of the individual columns:

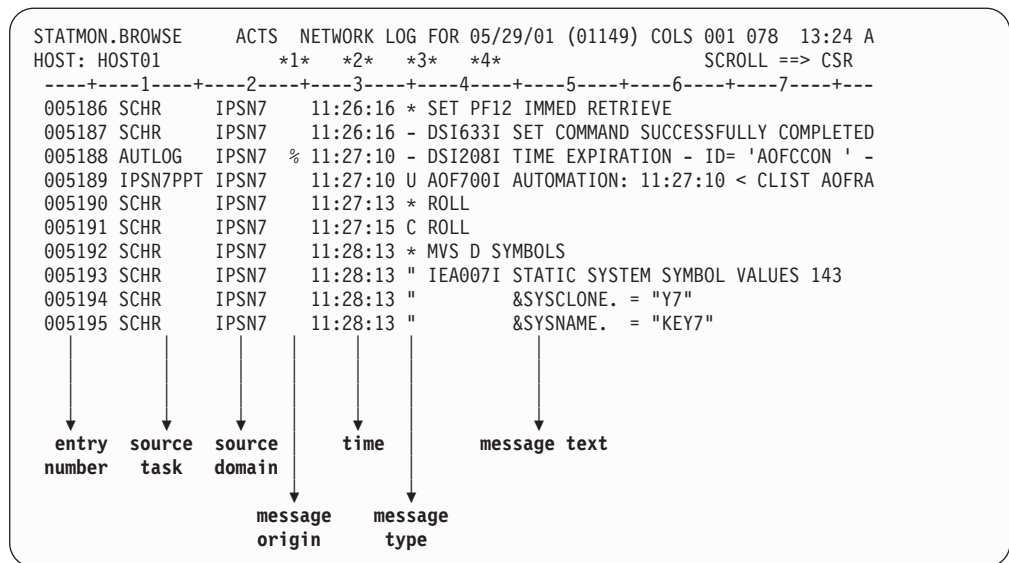


Figure 9. Network log entries

The following table explains the codes that can occur in the **Message Origin** column.

Table 7. Message origin codes

Code	Explanation
P	Message from the PPT task (VTAM)
%	Message to the authorized message receiver
P%	Message to the authorized receiver from the PPT
*	Message to a secondary receiver
P*	Message to a secondary receiver from the PPT
+	Message to a copied receiver (assigned with COPY)

The following table explains the more important codes that can occur in the **Message Type** column.

Table 8. Message type codes

Code	Explanation
C	Message or command generated by a CLIST
E	Message from the operating system interface
M	Message from a message command
Q	Unsolicited message from VTAM
R	Indicates that an operator entered the VTAM REPLY command in response to NetView WTOR number DSI802A. This message type is logged but does not appear on msys for Operations consoles.
S	Message text modified by user exit
t	Indicates a message response from a TSO command.
U	Message from locally-written programming
V	VTAM command entered from the system console
X	Indicates a cross-domain (NNT-OST) command.

Table 8. Message type codes (continued)

Code	Explanation
Y	VTAM message from the system console.
Z	Indicates a message from a data services task (DST).
!	Indicates a message from an immediate command processor. When displayed in the immediate message area on the screen, the message type and domain name are not displayed. When received cross-domain, this type of message is in the normal output area, along with its domain name and type prefix.
–	Message from NVSS
*	Command from a terminal operator (command echo)
+	Message from programs other than msys for Operations
>	Message requiring a reply
'	Multi-line message from msys for Operations
"	Multi-line message from MVS
=	Multi-line message from non-IBM code
	Indicates a message generated in a pipeline.

Log browse filtering

The BLOG command activates the netlog browse facility based on filters. You can select which records to display using any combination of the following filters:

- Select a local or remote msys for Operations. The default is the local msys for Operations. Changing the msys for Operations domain, Netid, or operid fields may result in browsing a remote msys for Operations log.
- Select the NETLOGA, NETLOGI, NETLOGS, or NETLOGP log.
- Select the starting display column.
- Select the operator ID for which records were logged.
- Select the origin domain of records that were logged.
- Select the message ID of messages that were logged.
- Select the starting time and date for records that were logged.
- Select the ending time and date for records that were logged.
- Select a character string that will be matched with the text of a message that was logged.

Chapter 12. Getting online help

To access the msys for Operations online help, enter `HELP item` on the command line, where *item* is the item for which you want help. This can be a command, a message ID, or an abend code. For the sysplex commands `INGCF` and `INGPLEX`, you can also specify the subcommand you are interested in. If you enter `HELP` without any additional parameter, a help menu for NVSS is displayed; note, however, that the `INGCF` and `INGPLEX` commands are not included in this menu.

Examples

To get help for the `DRAIN` subcommand of the Sysplex Functions command `INGCF`, enter:

```
HELP INGCF DRAIN
```

To get help for the NVSS `BROWSE` command, enter:

```
HELP BROWSE
```

To get help for message `CNM937I`, enter:

```
HELP CNM937I
```

To get help for a user abend code (`Uxxx`), enter:

```
HELP ABEND
```

Note: If you receive an msys for Operations message that includes an unexplained return code, see “Macro return codes” on page 217.

Part 4. Command reference

Chapter 13. General commands

This chapter describes the commands that are supplied by NVSS. For online help on a specific command, enter:

`HELP command`

where *command* is the name of the command.

ALL (BROWSE, WINDOW)

Purpose

Use the ALL command to display a specified collection of lines in BROWSE and WINDOW. If parameters are not specified, all lines are displayed and current filtering is disregarded. If */string/* is specified, only lines matching the string are displayed.

For more information, refer to the online help.

Examples

Displaying only lines containing characters TASK

To display only lines with characters TASK, enter the following command from the BROWSE or WINDOW command line:

`ALL /TASK/`

AUTOWRAP

Purpose

The AUTOWRAP command controls whether your terminal is held when the screen is full of data, or if the screen is automatically overlaid with new data.

For more information, refer to the online help.

Examples

Setting wrap display time

To set AUTOWRAP to display new data seven seconds after the screen is full, enter:

`AUTOWRAP 7`

BACK (BROWSE, HELP, WINDOW)

Purpose

The BACK command scrolls backward to the beginning of the data.

For more information, refer to the online help.

Examples

Displaying a help panel further than one panel back

If you want to navigate to a help panel that is three pages back, enter one of the following commands:

BACK 3

B 3

BLOG

Purpose

The BLOG command activates the log browse facility, showing a subset of the information based on filtering criteria.

When used with no parameters, the BLOG command will display a full-screen input panel where the filtering criteria can be entered. When used with parameters, the BLOG command will start the log browse facility based on the filtering criteria provided on the command line arguments.

For more information, refer to the online help.

Examples

Using the BLOG input panel

To use the BLOG input panel, enter:

BLOG

The BLOG input panel is displayed where you can enter filtering information. When you have entered your choices, press **Enter** to start the log browse facility.

BOTTOM (BROWSE, HELP)

Purpose

The BOTTOM command displays the last page of a multipage panel.

For more information, refer to the online help.

BROWSE

Purpose

The BROWSE command enables you to scan the netlog or members of a partitioned data set (PDS).

The members are contained in a partitioned data set.

For more information, refer to the online help.

Examples

The format of dates and times specified in the following examples assumes the default setting for date and time formats on the DEFAULTS and OVERRIDE commands.

Browsing the active netlog for a specified time

To browse the netlog (either primary or secondary) that is currently active from 2:40 p.m. on February 5, 2001 to 2:00 a.m. on February 6, 2001, enter:

```
BROWSE NETLOGA FROM 02/05/01 14:40 TO 02/06/01 2:00
```

Browsing the inactive netlog

To browse the netlog (either primary or secondary) that is currently inactive, enter:

```
BROWSE NETLOGI
```

Browsing a DSICMD member

When you want to browse a DSICMD member, but do not wish to have the included members resolved, enter:

```
BROWSE DSICMD NOINCL
```

DEFAULTS

Purpose

The DEFAULTS command sets msys for Operations-wide defaults.

You can override some of the DEFAULTS command settings for a specific operator ID using the OVERRIDE command.

You can use the LIST DEFAULTS command to get a list of the current DEFAULTS settings and the number of dumps that have been taken for storage overlay or control block overwrite conditions (DMPTAKEN).

For more information, refer to the online help.

Examples

Changing the banner

To change the banner on the logon and message display panels, enter one of the following:

```
DEFAULTS BANNER=OneBigWord
```

```
DEFAULTS BANNER='Up to 24 characters'
```

DISPFK

Purpose

The DISPFK command enables you to display or save the PF key settings.

For more information, refer to the online help.

END (BROWSE, HELP)

Purpose

The END command stops the current component panel sequence and returns to the component that was previously active.

For more information, refer to the online help.

FIND (BROWSE)

Purpose

The FIND command locates specific information while browsing a data set and a member. You can search for a previous entry or for the next entry. The default is NEXT. You can limit columns to be searched by specifying *left* and *right* column numbers.

For more information, refer to the online help.

Examples

Finding the next occurrence of a specified string

To find the next occurrence of DSI, enter:

```
FIND DSI
```

Or

```
F DSI
```

Finding the previous occurrence of a specified string

To scan the lines previous to the current line for an occurrence of the string DSIDMN, enter:

```
FIND DSIDMN PREV
```

Or

```
F DSIDMN P
```

FIND (WINDOW)

Purpose

The FIND command locates specific information while displaying data with the WINDOW command. This includes command and message help, and index information. You can search for a previous entry or for the next entry. The default is NEXT. You can limit columns to be searched by specifying *left* and *right* column numbers.

The search begins where the cursor is located, if the cursor is in the display. Otherwise, the search begins at the first line of information displayed on your screen.

For more information, refer to the online help.

Examples

Finding the next occurrence of a specified string that limits the search to specified columns

To find the next occurrence of DSIDMN in columns 1–90, enter:

```
F 'DSIDMN' 1 90
```

FORWARD

Purpose

The FORWARD command scrolls forward toward the end of the data.

For more information, refer to the online help.

Examples

Advancing a specified number of help panels

If you want to move ahead five help panels, enter:

```
FORWARD 5
```

HELP

Purpose

The HELP command displays help information for messages and commands.

You can use the following commands while you are using the HELP facility:

- BACK
- BOTTOM
- END
- FORWARD
- HELP
- RETURN
- TOP

For more information, refer to the online help.

Examples

Displaying help for commands

To receive help for the INGPLEX command, enter:

```
HELP INGPLEX
```

Displaying an online help panel for a specified message

To display an online help panel for message CNM937I, enter:

```
HELP CNM937I
```

The action suffix (I) is not required.

INDEX

Purpose

The INDEX command displays subjects that are explained in the online help facility. Use the backward and forward PF key to move through the index. Use the FIND command to search for a particular topic. If you find a topic you want to refer to, enter the option next to the topic.

For more information, refer to the online help.

Examples

Displaying index entries

To display the online index, enter:

```
INDEX
```

To display all index entries beginning with the letter R, enter:

```
INDEX R
```

INPUT

Purpose

The INPUT command modifies the length of the input area of the message display screen. The input area is at the bottom of the message display screen.

For more information, refer to the online help.

Examples

Changing the input area to a specified number of lines

To change the command entry area to two lines, enter:

```
INPUT 2
```

LIST

Purpose

The LIST command gives information about your msys for Operations session.

For more information, refer to the online help.

Examples

Displaying the current defaults

To display the current defaults, enter:

```
LIST DEFAULTS
```

For an explanation of the various defaults, see the DEFAULTS command. Note that some values, for example, SENDMSG, SCRNFMT, and SCROLL, are not valid on the DEFAULTS command.

LISTA

Purpose

The LISTA command displays the data set status, disposition, *ddnames*, and data set names of the files currently allocated to the msys for Operations program. It can also indicate which data sets contain a specific member.

The LISTA command lists the files allocated to the msys for Operations program. This includes files allocated through JCL and those allocated dynamically. In addition, (OPER-DS) indicates an operator data set designated by the OVERRIDE command. Also, (INSTORE-COMMON) indicates a member loaded by the INSTORE stage.

For more information, refer to the online help.

Examples

Listing a file with a specified ddname and member

The following example illustrates how to find the data set in DSIPARM that contains member DSITBL01:

```
LISTA DSIPARM DSITBL01
```

Listing all allocated files

To list all allocated files, enter:

```
LISTA
```

LOCATE (BROWSE)

Purpose

The LOCATE command enables you to position your log browse display to a given record number, or to a given date and time. When used with no parameters, the LOCATE command will position the log browse display to the first record of the date currently being displayed.

For more information, refer to the online help.

Examples

Locating 11 A.M. of today's date

To locate the log browse display at 11 a.m. of today's date, enter:

```
LOCATE 11:00 TODAY
```

LOGOFF

Purpose

The LOGOFF command ends the session between your terminal and the system. When your task terminates, some of your messages are rerouted to another authorized receiver. The messages rerouted include all those messages from VTAM or from your MVS operating system that require a reply or action from you, and any messages that were routed to you as a primary receiver but were not processed at the time of termination.

For more information, refer to the online help.

MSG

Purpose

The MSG command sends a message to an operator or to the netlog.

For more information, refer to the online help.

Examples

Sending a message to all active terminals and system console operators

To send a message indicating system shutdown to all active terminals and to the system console operator, enter:

```
MSG ALL,SYSTEM SHUTDOWN IN 15 MINUTES
```

MVS

Purpose

The MVS command enables you to enter an MVS system operator command from thesys for Operations program. If your task has not obtained an MVS console, the MVS command attempts to obtain one for you.

For more information, refer to the online help.

Examples

Using the MODIFY command

To use the MODIFY command, enter:

```
MVS MODIFY TSO,USERMAX=nnnn
```

OVERRIDE

Purpose

The OVERRIDE command can be used to specify options for a particular operator. The OVERRIDE options take precedence over the options specified by the DEFAULTS command. The OVERRIDE options that are related to message display (BEEP, DISPLAY, HOLD) apply to all messages that are to be displayed at the individual operator's terminal.

Use the LIST OVERRIDE command to request a list of the current OVERRIDE settings.

For more information, refer to the online help.

Examples

Changing the command priority

To change the command priority, enter:

```
OVERRIDE CMD=low
```

REPEAT (BROWSE)

Purpose

The REPEAT command reissues the last FIND command while you are browsing the netlog or a member of a partitioned data set. Because the REPEAT command is sensitive to the current position of the cursor, it is normally entered using a PF key.

By repeatedly pressing the PF key set to REPEAT, you can find successive occurrences of a specified character string. When the first occurrence of a character

string has been found, the REPEAT key will find the next occurrence. When the last occurrence of a character string has been found, the REPEAT key can be used to continue the search, wrapping around from the bottom line to the top line (or from the top line to the bottom line if the FIND command included the PREV parameter).

For more information, refer to the online help.

RETRIEVE

Purpose

The RETRIEVE command places the last command you issued in the command input area. This command gives you a convenient method to review, rerun, or edit and rerun commands you have recently entered.

For more information, refer to the online help.

RETURN (BROWSE, HELP)

Purpose

The RETURN command returns you to the previous component or the last selection panel that you used.

For more information, refer to the online help.

ROLL

Purpose

The ROLL command returns to a previous component and the last panel that you used in that component.

The system remembers the sequence in which you go from one component to another. When you use the ROLL command, the system moves the name of your current component to the beginning of the sequence of components, and brings up the component at the end of the sequence, displaying the panel that was displayed when you left that component.

For more information, refer to the online help.

SET

Purpose

The SET command defines PA and PF keys for the command facility or a full-screen application that supports its PF or PA settings. These settings remain valid until you delete them or log off.

For more information, refer to the online help.

Note: The PF key settings for the INGPlex and INGCF commands cannot be changed with the SET command.

Examples

Setting PF12 to retrieve your last command

To set the PF12 key to retrieve your last command, enter:

```
SET PF12 IMMED RETRIEVE
```

TOP (BROWSE, HELP)

Purpose

The TOP command displays the first page of a multipage panel.

For more information, refer to the online help.

WHO

Purpose

The WHO command list displays the status of all operator terminals and information about your session.

After entering the WHO command list, you see information similar to the following:

```
* MS001   WHO
C MS001   LIST STATUS=OPS
- MS001   OPERATOR: OPER1      TERM: A01A701  STATUS: ACTIVE
- MS001   OPERATOR: AUTO1      TERM: AUTO1    STATUS: ACTIVE
- MS001   OPERATOR: AUTO2      TERM: AUTO2    STATUS: ACTIVE
- MS001   END OF STATUS DISPLAY
C MS001   LIST STATUS=NNT
- MS001   MAX SESS: 00000005
- MS001   NO ACTIVE NCCF TO NCCF SESSIONS FOUND
C MS001   LIST OPER1
- MS001   STATION: OPER1      TERM: A01A701
- MS001   HCOPY: NOT ACTIVE  PROFILE: DSIPROFA
- MS001   STATUS: ACTIVE
- MS001   AUTHRCVR: NO       CONTROL: GLOBAL
- MS001   OP CLASS LIST: 2
- MS001   DOMAIN LIST: MS001 (I) MS002 (I) MS003 (I)
- MS001   ACTIVE SPAN LIST: NONE
- MS001   END OF STATUS DISPLAY
```

For more information, refer to the online help.

WINDOW

Purpose

The WINDOW command is a full-screen application that captures and displays data from other commands that would normally display messages. The WINDOW command facilitates searching the captured data and enables you to scroll forward and back, as well as left and right. WINDOW is also ROLLable.

For more information, refer to the online help.

Examples

Displaying a data set wider than 80 characters

To display member XYZ of data set USER.LISTING which is greater than 80 characters wide, enter:

```
WINDOW < 'USER.LISTING(XYZ)'
```

Chapter 14. Sysplex-related commands

This section contains reference information about the INGCF and INGPlex commands, which support several actions. Some of these actions impact the system configuration. Others only serve to display information. The display actions are accessible for every operator who can call the respective command. Access to related groups of actions that modify the system configuration can be granted or denied to operators individually.

Note: The actions that are controlled by this security mechanism are marked by an asterisk (*) in the following descriptions.

The layout of the panels from which you initiate an action depend on your authorizations. The code or PF key by which you initiate a certain action is only displayed when you are authorized to perform the action.

Purpose

The INGCF command supports all the functions of msys for Operations that deal with coupling facilities. It supports full mode and line mode. If you issue INGCF in line mode, only the display functions are available. Therefore, you cannot start an action in msys for Operations when you issue INGCF from an MVS console.

The INGCF command supports the following parameters:

- **DRAIN**
Serves to remove all allocated structures from the coupling facility, to disconnect the coupling facility from the systems of the sysplex, and to inactivate the coupling facility.
- **ENABLE**
Serves to activate a coupling facility, connect it with the systems of a sysplex and to populate it with structures.
- **PATH**
Displays and controls the sender paths of the target coupling facility. It sets the sender paths ONLINE and OFFLINE physically and logically.
- **STRUCTURE**
Displays detail information and rebuilds or deletes a selected structure on the target coupling facility. It also lets you start and stop duplexing.

INGCF associates a status with every coupling facility, and a condition with every structure (instance) that is allocated on the target coupling facility. The structure condition is influenced by the release level of the system that allocated the structure. The INGCF functions use the coupling facility state and the structure conditions to determine which action can be performed in any given situation. Therefore, the DRAIN and ENABLE functions can enforce a correct sequence of actions for complex tasks such as draining or restoring a coupling facility.

If the selected action impacts the sysplex configuration it must be confirmed before execution.

Authorizations

The actions you can initiate with INGCF depend on your authorizations. You can see your authorization type on the panels. Note that the authorization types refer to the current function, and that your authorization type can be different for different functions.

The following authorization types exist:

DISPLAY

You cannot initiate any action that affects the sysplex configuration.

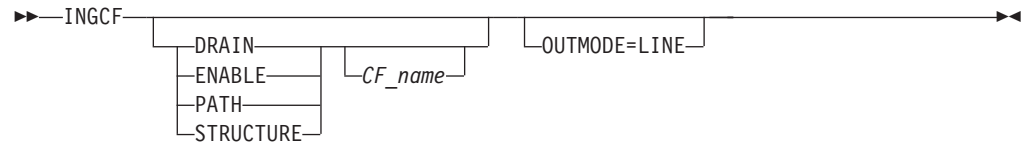
ALL BUT (ACTIVATE|SHUTDOWN)

This type only occurs in the DRAIN and ENABLE command dialogs. You can rebuild structures, force the deletion of structures and set the sender paths offline and online, but you cannot inactivate or activate the coupling facility.

ALL You can initiate all actions from the corresponding panel.

Depending on your authorizations, it is possible that you have, for example, authorization type ALL for the STRUCTURE function, and authorization type DISPLAY for the DRAIN function.

Syntax



Parameters

DRAIN

Prepares a coupling facility for removal from the sysplex.

ENABLE

(Re)integrates a coupling facility into a sysplex.

PATH

Displays the sender paths of a coupling facility.

STRUCTURE

Offers manipulation of individual structures (detail information, rebuild, deletion).

CF_name

Name of the target coupling facility for the specified function. The default is a selection panel that shows all available coupling facilities of the sysplex.

OUTMODE

If you specify OUTMODE=LINE, INGCF is called in line mode. In this mode, only the display functions of the command are available.

Restrictions and Limitations

The ENABLE and the PATH functions require that the active IODF are catalogued. Otherwise, sender path information cannot be retrieved in certain situations.

INGCF ENABLE assumes that the receiver paths from the coupling facility to the systems of the sysplex have been defined and activated. This requires a POR of the CPC on which the coupling facility resides.

Coupling Facility States

The status of a coupling facility can be as follows:

ACTIVATING

The coupling facility is being activated and will then become DRAINED.

DEACTIVATING

The coupling facility is being deactivated and will then become INACTIVE.

DRAINING

The coupling facility is being disconnected from the connected systems.

DRAINED

The coupling facility has no connection to any system and can be removed from the sysplex.

DRAINED NOHWACC

The coupling facility has no connection to any system, but cannot be removed from the sysplex because the BCP (Basic Control Program) internal interface is not available.

Note: This status is also displayed when the coupling facility has been deactivated from the HMC (Hardware Management Console) but the XCF display commands still return the name of the coupling facility.

ENABLING

The coupling facility is being connected to the systems of the sysplex that use it.

FORCING

Allocated structures are being deleted from the coupling facility. This only happens with structures that have no active connectors, and with these only when they cannot be rebuilt by system-managed rebuild.

INACTIVE

The coupling facility is not active.

INACTIVE NOHWACC

The coupling facility is not active and cannot be activated because the BCP internal interface is not available.

NORMAL

The coupling facility may have allocated structures and is connected to all systems.

NORMAL OFFLINE

The coupling facility may have allocated structures. At least one system has set all its sender paths to this coupling facility to OFFLINE. XES will reject any rebuild request for this coupling facility.

NOTINPOLICY

The coupling facility is active but not defined in the active CFRM policy.

POPULATING

The coupling facility is being populated with all those structures that have it on the first place in their preference list.

REBUILDING

Either all allocated structures that can be rebuilt are being removed from the coupling facility by the XES rebuild process (initiated by DRAIN), or one particular such structure is being removed (initiated by the STRUCTURE).

Structure conditions

The condition of an allocated structure can be:

Rebuild is not supported.

The structure can neither be rebuilt, nor can its deletion be forced.

The structure has at least one active connector that does not support user-managed rebuild, and at least one active connector that does not support system-managed rebuild.

System-managed processes not supported.

The structure cannot be rebuilt, nor can its deletion be forced.

System-managed rebuild, which is a system-managed process, is not possible for one of the following reasons:

- The structure was allocated from a system with OS/390 2.7 or below.
- The CFRM couple data sets have not been formatted to support system-managed processes (ITEM NAME(SMREBLD) NUMBER(1) was not specified.).

Note: In certain rare cases system-managed processes are not supported although the condition that is displayed on the DRAIN panel seems to indicate the contrary. Then, the rebuild will be initiated, but will fail with message IXC367I indicating that system-managed processes are not supported for the structure.

No alternate coupling facility defined or available.

The structure can neither be rebuilt, nor can its deletion be forced.

The structure has an active connector and supports rebuild but has no alternate coupling facility defined in its preference list, or the alternate coupling facilities defined in the preference list are currently unavailable.

Insufficient space detected for rebuild.

The structure can or could not be rebuilt. Its deletion cannot be forced.

No alternate coupling facility has enough space to rebuild the structure.

Preference list is empty.

The structure cannot be rebuilt because its preference list is currently empty. A possible reason for this is a pending policy change; for pending policy changes, see "P column" on page 123.

Structure is pending deallocation.

XES accepted a forced deletion of the structure but does the real deallocation later.

Note: This status can only occur when MVS APAR OW39404 has not been installed.

Structure is being rebuilt.

The structure is being rebuilt to another coupling facility.

Duplex rebuild is being stopped.

Two instances of the structure were maintained on different coupling facilities. The application is being disconnected from that instance that is allocated on the target coupling facility. After disconnecting, the instance is deleted.

No connection exists.

The structure cannot be rebuilt, but you can force its deletion.

The structure has no connections and cannot be rebuilt with system-managed rebuild.

No alternate coupling facility for structure with no connections.

The structure cannot be rebuilt, but you can force its deletion.

The structure has no connections. It could be rebuilt with system-managed rebuild, but no alternate coupling facility is defined in its preference list or available.

No alternate coupling facility for structure with no active connections.

The structure cannot be rebuilt, but you can force its deletion.

The structure has only DISCONNECTING, FAILED, or FAILED-PERSISTENT connections. It could be rebuilt with system-managed rebuild, but no alternate coupling facility is defined in its preference list or available.

The structure's initial size is less than its actual size.

The structure can be rebuilt, but this can lead to loss of data.

An initial size is specified for the structure in the active CFRM policy. This initial size was used for the allocation of the structure. Afterwards, the size of the structure was increased either by the application itself or an operator command. However, the structure will only be rebuilt with its initial size. Subsequently, INGCf will expand the structure to its actual size again, but this will happen *after* the data have been transferred. To avoid a potential loss of data, the application has to change the initial size to the actual size.

No active connection exists.

The structure cannot be rebuilt, but you can force its deletion.

The structure has only DISCONNECTING, FAILED, or FAILED-PERSISTENT connections and cannot be rebuilt with system-managed rebuild.

Note:

INGCF DRAIN deallocates structures with this condition as part of the REBUILD action (see "REBUILD (F10)" on page 112). INGCf STRUCTURE accepts a rebuild request for structures with this condition, but deallocates them (see "Rebuild (R)" on page 122).

No connections. System-managed rebuild supported.

The structure can be rebuilt.

The structure has no connections, but can be rebuilt with system-managed rebuild.

No active connections. System-managed rebuild supported.

The structure can be rebuilt with system-managed rebuild.

User-managed rebuild is not possible for the structure because it has only DISCONNECTING, FAILED, or FAILED-PERSISTENT connections.

System-managed rebuild is supported

The structure can be rebuilt.

The structure has active connectors. At least one active connector does not support user-managed rebuild, but all active connectors support system-managed rebuild.

Duplex rebuild is active.

The application is connected to two instances of the same structure on different coupling facilities.

[No condition]

When no condition is displayed, the structure can be rebuilt.

The structure has at least one active connection, and all its active connectors support user-managed rebuild.

Structure is awaiting rebuild.

The structure has been selected for rebuild but has not been processed yet.

Structure is currently allocated on *cf_name*.

The structure can be rebuilt on the target coupling facility with the POPULATE action of the ENABLE function. It is currently allocated on the *cf_name* coupling facility, but the target coupling facility precedes *cf_name* in the structure's preference list. This condition is displayed only in the ENABLE command dialog.

Structure allocated in *cf_name* cannot be rebuilt to this CF.

The structure can probably not be rebuilt on the target CF with the POPULATE action of the ENABLE function. It is currently allocated in the *cf_name* CF, but the target CF precedes *cf_name* in the structure's preference list. And, the actual size of the structure is greater than the free space of the target CF. This condition is displayed only in the ENABLE command dialog.

Example

If you issue INGCF without any parameters, a panel with all coupling facilities of the sysplex is displayed:

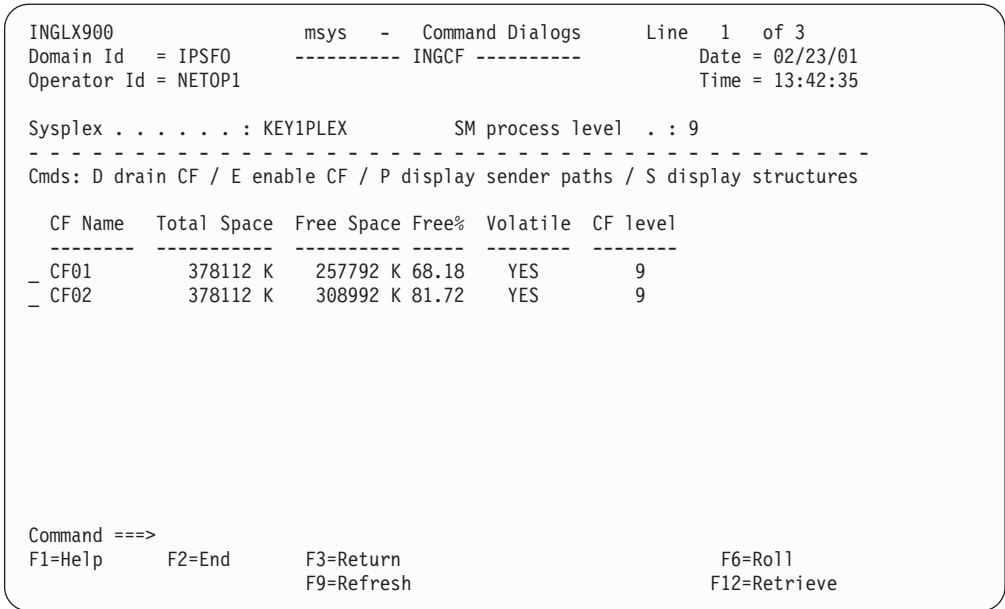


Figure 10. INGCF selection panel

Specify a function for a selected coupling facility and press ENTER.

DRAIN

Purpose

The DRAIN function of INGCF facilitates the removal of a coupling facility from the sysplex, for example, for maintenance purposes. With this option, you can perform the following sequence of tasks:

1. Display information for all allocated structures of the coupling facility.
2. Rebuild all rebuildable structures on *another* coupling facility, and delete instances of structures on the target coupling facility that are being duplexed on another coupling facility.

INGCF DRAIN

Notes:

- a. The scope of the structures that can be rebuilt depends on the release level of the sysplex members.
 - b. INGCF DRAIN rebuilds structures one at a time (SETXCF START,REBUILD,STRNAME=), not globally (SETXCF START,REBUILD,CFNAME=), and always on a coupling facility that is different from the target coupling facility (LOCATION=OTHER).
 - c. Generally, you should be aware that it is XES that performs the actual rebuild. Not all of the factors that XES takes into account when allocating a structure are accessible to msys for Operations. Therefore, a rebuild request for a structure that should be rebuildable according to its condition can fail in certain rare cases.
3. Force the deletion of structures that have no active connectors and cannot be rebuilt.

Note that there are structures that you can neither rebuild nor delete with the force action. These include the structures that have at least one active connector and do not support rebuild. To remove such structures first disconnect all active connectors, and then delete the structure manually if it is persistent or has persistent connections.

4. Disconnect the coupling facility from the systems with which it is connected.
5. Inactivate the target coupling facility.

INGCF DRAIN ensures that these actions are performed in the correct order, as specified above.

Actions

The following F-keys are supported:

*REBUILD (F10)

Starts the rebuild of structures that can be rebuilt on *another* coupling facility. Thus, a rebuild is only initiated for structures whose preference list contains more than one coupling facility.

There are two methods for rebuild, user-managed and system-managed rebuild. User-managed rebuild is supported for all release levels. System-managed rebuild is only available with systems that have OS/390 2.8 or above; it must have been enabled by formatting the CFRM couple data sets with the specification

ITEM NAME(SMREBLD) NUMBER(1)

System-managed rebuild is only performed when the requirements for user-managed rebuild are not met. This applies, for example, to structures without active connectors.

The REBUILD action also deletes all structure instances on the target coupling facility that are being duplexed on another coupling facility.

Note:

The REBUILD action *deallocates* structures with the condition 'No active connection exists.'. See "No active connection exists" on page 110.

*FORCE (F5)

Forces the deallocation of structures with one of the following conditions:

- No connection exists.

- No alternate coupling facility for structure with no active connections.
- No alternate coupling facility for structure with no connections.

This action is only made available after all structures that can be rebuilt have been rebuilt.

***DRAIN (F4)**

Disconnects the coupling facility from its connected systems by setting the sender path(s) OFFLINE.

This action is only enabled after all structures of the target coupling facility have been removed to another coupling facility or deallocated. Note that structures that have active connectors but do not support rebuild cannot be removed with F10 or F5. They must be deallocated manually before executing this step is enabled.

***SHUTDOWN (F11)**

This action inactivates the coupling facility. It is only made available when all connections between the coupling facility and the systems of the sysplex have been disconnected.

Note that these actions can only be performed if INGC F DRAIN is issued in full mode. In line mode, only the display function is available.

To avoid performance degradation due to multiple rebuild processes, or unpredictable results due to multiple executions of an action, all actions are locked. Therefore, an action is rejected if any lock exists even if the action does not affect the action currently being performed. Because the action can take a long time it is also executed asynchronously on a dedicated autotask, preventing the operator from being blocked. To check progress, use the refresh function (F9).

Example

In the following example, a coupling facility is drained:

1. All of its structures that can be rebuilt are rebuilt on another coupling facility, and duplexing is stopped.
2. For all structures that have no active connector and cannot be rebuilt deletion is forced.
3. All systems that are connected with the coupling facility are disconnected.
4. The coupling facility is inactivated.

When you issue INGC F with the option DRAIN, you can specify the coupling facility to be drained, for example by entering INGC F DRAIN CF01; in this case, the panel of Figure 11 on page 114 is displayed at once. If you do not specify a coupling facility name, INGC F displays a selection panel with all coupling facilities that are defined in the sysplex. After selection of CF01, INGC F displays the following panel:

INGCF DRAIN

```

INGLX901          msys - Command Dialogs      Line 1 of 31
Domain Id   = IPSFO ----- INGCF DRAIN ----- Date = 02/13/01
Operator Id = NETOP1                               Time = 07:01:00

Coupling Facility ==> CF01          Status . . . . . : NORMAL
Sysplex . . . . . ==> KEY1PLEX      Permission . . . . . : ALL
-----
Structure          Condition
-----
DFHXQLS_TESTTSQ1  No active connections. System-managed rebuild supported.
DSNG_LOCK1
DSNG_SCA           System-managed rebuild is supported.
ISGLOCK
ISTGENERIC
IXCGRS
IXCplex_PATH1
M7SG_LOCK1        *No alternate CF for structure with no active connections.
M7SGEMHQ          No active connections. System-managed rebuild supported.
M7SGMSGQ          No active connections. System-managed rebuild supported.
M7SGMSGQOV        No active connections. System-managed rebuild supported.

Command ==>
F1=Help           F2=End           F3=Return          F6=Roll
                  F8=Forward      F9=Refresh      F10=Rebuild       F12=Retrieve

```

Figure 11. DRAIN command dialog: panel before any action

The status of the coupling facility (NORMAL) and the authorization type of the operator (ALL) are displayed on the right side of the panel header. The main part of the panel consists of a list containing the structures allocated in CF1 and their conditions. The conditions are classified by color and an asterisk. The asterisk signifies that a structure cannot be rebuilt.

The only action that is enabled is REBUILD with F10. Pressing F10 calls the following confirmation panel:

```

INGLX92R          msys - Command Dialogs      Date = 02/13/01
Domain Id   = IPSFO ----- INGCF DRAIN ----- Time = 07:01:04
Operator Id = NETOP1

Coupling Facility . : CF01
Sysplex . . . . . : KEY1PLEX

                        R E B U I L D Confirmation

The REBUILD process runs asynchronously on the next system in the sysplex that
has access to the CFRM couple data set and can perform all necessary actions.
Each structure that has no * indicator in front of its status is rebuilt to its
status accordingly. The structures are processed in sequence. Once started use
the refresh PF key for getting the current status of the process. When more
than one structure is being rebuilt a timeout occurred indicating that XCF is
very busy. But processing continues. A display without any structure or only
structures that cannot be rebuilt indicates a successful completion.

Command ==>
                  F2=End           F3=Return          F6=Roll
                  F10=Go           F11=Cancel        F12=Retrieve

```

Figure 12. DRAIN command dialog: confirmation panel for REBUILD

After F10 was pressed and the rebuild is complete the command dialog can be refreshed with F9. It looks as follows:

```

INGLX901      msys - Command Dialogs      Line 1 of 1
Domain Id    = IPSFO      ----- INGCF DRAIN -----      Date = 02/13/01
Operator Id  = NETOP1      Time = 07:53:36

Coupling Facility ==> CF01      Status . . . . . : NORMAL
Sysplex . . . . . ==> KEY1PLEX      Permission . . . . . : ALL
-----
Structure      Condition
-----
M7SG_LOCK1      *No alternate CF for structure with no active connections.

Command ==>
F1=Help      F2=End      F3=Return      F5=Force      F6=Roll
F9=Refresh      F12=Retrieve

```

Figure 13. DRAIN command dialog: panel after rebuild

One structure could not be rebuilt because no alternate coupling facility is specified in its preference list. The REBUILD action is no longer available. Instead, the FORCE action (F5) is available because the structure that could not be rebuilt has a condition that allows forcing the deallocation of the structure. Pressing F5 calls a confirmation panel similar to that for REBUILD. Pressing F10 on the confirmation panel and refreshing the command dialog after the action has been completed results in the following panel:

```

INGLX901      msys - Command Dialogs      Line
Domain Id    = IPSFO      ----- INGCF DRAIN -----      Date = 02/13/01
Operator Id  = NETOP1      Time = 08:12:28

Coupling Facility ==> CF01      Status . . . . . : NORMAL
Sysplex . . . . . ==> KEY1PLEX      Permission . . . . . : ALL
-----
Structure      Condition
-----

Command ==>
F1=Help      F2=End      F3=Return      F4=Drain      F6=Roll
F9=Refresh      F12=Retrieve

```

Figure 14. DRAIN command dialog: panel after forcing

No more structures are allocated in the coupling facility, so that the coupling facility can be released from the connections with the systems of the sysplex. Consequently, INGCF DRAIN enables the DRAIN action (F4). After completion of

INGCF DRAIN

that action, the status of the coupling facility changes to DRAINED, as shown on the following panel:

```
INGLX901      msys - Command Dialogs      Line
Domain Id   = IPSFO      ----- INGCF DRAIN -----      Date = 02/13/01
Operator Id = NETOP1                                           Time = 08:12:32

Coupling Facility ==> CF01      Status . . . . . : DRAINED
Sysplex . . . . . ==> KEY1PLEX      Permission . . . . . : ALL
-----
Structure      Condition
-----

Command ==>
F1=Help      F2=End      F3=Return      F6=Ro11
F9=Refresh      F11=Shutdown      F12=Retrieve
```

Figure 15. DRAIN command dialog panel: panel after draining

Because the coupling facility is no longer connected to any system, it can be inactivated. After pressing F11 the status of the coupling facility changes to INACTIVE:

```
INGLX901      msys - Command Dialogs      Line
Domain Id   = IPSFO      ----- INGCF DRAIN -----      Date = 02/13/01
Operator Id = NETOP1                                           Time = 08:12:32

Coupling Facility ==> CF01      Status . . . . . : INACTIVE
Sysplex . . . . . ==> KEY1PLEX      Permission . . . . . : ALL
-----
Structure      Condition
-----

Command ==>
F1=Help      F2=End      F3=Return      F6=Ro11
F9=Refresh      F12=Retrieve
```

Figure 16. DRAIN Command Dialog Panel: Panel After Inactivation

ENABLE

Purpose

The ENABLE function of the INGCF command is intended to support the (re)integration of a coupling facility into a sysplex. With this option, you can:

1. Activate the target coupling facility.

2. Connect the systems of the sysplex with the coupling facility.
3. Switch to another CFRM policy if the target coupling facility is not defined in the active policy and a suitable policy is available.

A suitable CFRM policy must contain:

- A definition of the target coupling facility
 - Appropriate definitions for every active coupling facility and every allocated structure
4. Rebuild all structures on the target coupling facility whose preference list starts with this coupling facility, unless this is excluded by other requirements.

INGCF ENABLE ensures that these actions are performed in the correct order, as specified above.

Actions

The possible actions and the associated F-keys are:

*ACTIVATE (F11)

This action activates the CFCC (Coupling Facility Control Code) through the BCP internal interface by an ACTIVATE command. At the end of the CFCC initialization the sender path(s) of all systems are set to ONLINE

*ENABLE (F4)

Sets the sender path(s) of all systems of the sysplex to ONLINE. This action is enabled when the coupling facility is active.

*SWITCH (F5)

Switches to another CFRM policy when the target coupling facility is not defined in the active CFRM policy and a suitable policy is available. When there is more than one suitable policy you can choose one of these from a selection panel.

A CFRM policy is suitable when it contains:

- A definition of the target coupling facility
- Definitions for every active coupling facility and every allocated structure

This action is only made available when the target coupling facility is active, but not defined in the current CFRM policy.

*POPULATE (F10)

Starts a rebuild process by which all structures that have the target coupling facility at the beginning of their preference list but are currently allocated on another coupling facility are allocated on the target coupling facility.

This action requires that the coupling facility be enabled, connected to all members of the sysplex, and defined in the current CFRM policy. The action is offered whenever INGCf ENABLE detects that a structure is not allocated on the target coupling facility although it is the preferred coupling facility of that structure.

Note: When you have drained a coupling facility with INGCf DRAIN and then reintegrate it with INGCf ENABLE, be aware that the set of structures that are allocated on the target coupling facility after population can be different from the original set before the draining. Typically, this happens when the original set does not contain


```

INGLX92E          msys - Command Dialogs
Domain Id   = IPSFO  ----- INCGF ENABLE -----
Operator Id = NETOP1                               Date = 02/20/01
                                                    Time = 11:06:20

Coupling Facility . : CF02
Sysplex . . . . . : KEY1PLEX

                E N A B L E  Confirmation

The ENABLE process runs asynchronously on the next system in the Sysplex that
has access to the CFRM couple data set. All sender paths of all system in the
sysplex are set to ONLINE. Once started use the refresh PF key for getting
the current status of the process. The status NORMAL indicates a successful
completion.

Command ==>
                F2=End          F3=Return          F6=Roll
                                F10=Go             F11=Cancel    F12=Retrieve

```

Figure 18. Confirmation panel for ENABLE

After pressing F10 on the confirmation panel, the command dialog changes as follows:

```

INGLX901          msys - Command Dialogs          Line 1 of 3
Domain Id   = IPSFO  ----- INCGF ENABLE -----          Date = 02/20/01
Operator Id = NETOP1                               Time = 11:06:39

Coupling Facility ==> CF02          Status . . . . . : NORMAL
Sysplex . . . . . ==> KEY1PLEX      Permission . . . . : ALL
-----
Structure      Condition
-----
GRPYCSQ_ADMIN  Structure is currently allocated in CF01.
GRPYHSAQUEUE   Structure is currently allocated in CF01.
HSA_LOG        Structure is currently allocated in CF01.

Command ==>
F1=Help      F2=End          F3=Return          F6=Roll
              F9=Refresh    F10=Populate      F12=Retrieve

```

Figure 19. ENABLE command dialog: panel after enabling

The status has changed to NORMAL, and F10 is enabled for populating the coupling facility. This implies that the target coupling facility is defined in the active CFRM policy.

The structure list contains three entries with the condition 'Structure is currently allocated in CF01.'. These are the structures that are currently allocated in CF01, but have CF02 at the first position in their preference list.

Pressing F10 populates the coupling facility, and the refreshed panel looks as follows:

INGCF ENABLE

```
INGLX901          msys - Command Dialogs      Line 1 of 3
Domain Id   = IPSFO      ----- INGCF ENABLE -----      Date = 02/20/01
Operator Id = NETOP1                                           Time = 11:17:35

Coupling Facility ==> CF02          Status . . . . . : NORMAL
Sysplex . . . . . ==> KEY1PLEX      Permission . . . . : ALL
-----
Structure      Condition
-----
GRPYCSQ_ADMIN   System-managed rebuild is supported.
GRPYHSAQUEUE    System-managed rebuild is supported.
HSA_LOG

Command ==>
F1=Help      F2=End      F3=Return      F6=Roll
              F9=Refresh      F12=Retrieve
```

Figure 20. ENABLE command dialog panel: panel after populating

The POPULATE action is no longer available because all structures whose preference list starts with CF02 are allocated in CF02.

PATH

Purpose

The INGCF PATH function displays the sender paths, that is, the paths from the connected systems to the specified coupling facility.

Example

```
INGLX903          msys - Command Dialogs      Line 1 of 4
Domain Id   = IPSFN      ----- INGCF PATH -----      Date = 06/20/01
Operator ID = HIR                                           Time = 10:28:49

Coupling Facility ==> CF01          Allocated Structures: 37
Sysplex . . . . . ==> KEY1PLEX      Permission . . . . : ALL
-----
Cmds: F set OFFLINE / N set ONLINE

  System  CHPID  Physical      Logical  Type
  -----
- KEY1    A5    ONLINE        ONLINE   CFS
-         A9    ONLINE        ONLINE   CFS
- KEY2    A5    ONLINE        ONLINE   CFS
-         A9    ONLINE        ONLINE   CFS
- KEY3    A5    ONLINE        ONLINE   CFS
-         A9    ONLINE        ONLINE   CFS
- KEY4    05    ONLINE        ONLINE   CFS
- KEY6    A5    ONLINE        ONLINE   CFS
-         A9    ONLINE        ONLINE   CFS
-

Command ==>
F1=Help      F2=End      F3=Return      F6=Roll
              F8=Forward  F9=Refresh      F12=Retrieve
```

Figure 21. The PATH dialog panel

The following command codes are available:

F Sets the sender path OFFLINE.

N Sets the sender path ONLINE.

- If you have issued INGCF with the PATH parameter, the Coupling Facility field is an input field. To display the path list of another coupling facility specify the name of the coupling facility in this field and press ENTER.
- The Allocated Structures field shows the number of allocated structures.
- The Permission field shows your authorization level.
- The System column contains the names of the systems that are connected to the target coupling facility.
- The CHPID column shows the IDs of the sender channel paths.
- The Physical column shows the status of the sender channel paths.
- The Logical column shows the logical status of the paths to that coupling facility.
- The Type column shows the type of the sender channel paths.

STRUCTURE

Purpose

The STRUCTURE function of the INGCF displays the allocated structures of a coupling facility. You can initiate a rebuild or deallocation of a selected structure if the conditions for these actions are satisfied.

Example

```

INGLX904          msys - Command Dialogs          Line 1 of 3
Domain ID = IPSFM  ----- INGCF STRUCTURE -----      Date = 02/22/02
Operator ID = NETOP1                               Time = 16:09:04

Coupling Facility ==> CF01
Sysplex . . . . . ==> KEY1PLEX          Permission . . . . : ALL
Include condition ==> YES (Yes/No - Condition retrieval takes longer)
-----
Cmds: D display details / F force / P stop duplex / R rebuild / S start duplex

  Structure      P D Condition
  -----
- DSNB_GBP0      U Duplex rebuild is active.
- DSNB_GBP32K    P U
- DSNB_LOCK1     S Duplex rebuild is active.
- DSNB_SCA       S System-managed rebuild is supported.
- ISGLOCK
- ISTGENERIC     System-managed rebuild is supported.
- IXCGRS
- IXCRLF
- M8SGEMHQ       System-managed rebuild is supported.
- M8SGMSGSL
- M8SGMSGQ       System-managed rebuild is supported.
- RRSARCH
- RRSDELAY
- RRSMAIN
- RRSREST        P *No alternate CF defined or available.

Command ==>
F1=Help      F2=End      F3=Return      F6=Roll
              F9=Refresh      F12=Retrieve

```

Figure 22. STRUCTURE dialog panel

The following actions codes are available:

D Displays detail information about the structure.

***F** Forces the deallocation of the structure if it has one of the following conditions:

INGCF STRUCTURE

- No connection exists.
- No alternate CF for structure with no active connections.
- No alternate CF for structure with no connections.

When you try to force the deallocation of a structure that can be rebuilt, an error message is issued.

***P** Stops duplexing of the selected structure.

***R** Starts the rebuild of the selected structure on another coupling facility. Depending on the PENDING status, the automation starts the rebuild to the same CF, or to another CF. A rebuild is only initiated for structures whose preference list contains more than one coupling facility.

There are two methods for rebuild, user-managed and system-managed rebuild. User-managed rebuild is supported for all release levels. System-managed rebuild is only available with systems that have OS/390 2.8 and above (or z/OS 1.2 and above); it must have been enabled by formatting the CFRM couple data sets with the specification

ITEM NAME(SMREBLD) NUMBER(1)

System-managed rebuild is only performed when the requirements for user-managed rebuild are not met. This applies, for example, to structures without active connectors.

INGCF STRUCTURE accepts a rebuild request for structures with the condition 'No active connection exists.', but *deallocates* them. See "No active connection exists" on page 110.

The rebuild function works differently depending on the status of the structure (PENDING calls LOCATION=NORMAL, otherwise LOCATION=OTHER).

***S** Starts duplexing of the selected structure.

There are two methods for duplexing, user-managed and system-managed duplexing. User-managed duplexing is supported for all release levels. System-managed duplexing is only available with systems that have OS/390 2.8 and above (or z/OS 1.2 and above); it must have been enabled by formatting the CFRM couple data sets with the specification

ITEM NAME(SMDUPLEX) NUMBER(1)

System-managed duplexing is only performed when the requirements for user-managed duplexing are not met. This applies, for example, to structures without active connectors.

Starting the duplex rebuild of a structure requires at least the policy entry allowing the duplex rebuild of the structure. If there is no entry the duplex rebuild is disabled. The other requirements depend on the type of the duplex rebuild. When all connectors to a structure allow user-managed duplex rebuild, this type takes precedence over system-managed duplex rebuild. However, user-managed rebuild also requires at least one active connector. Thus, when the operator starts the duplex rebuild for a structure allowing user-managed duplex rebuild as well as system-managed rebuild but without having active connectors, XCF tries to initiate a system-managed duplex rebuild. System-managed duplex rebuild has the following requirements:

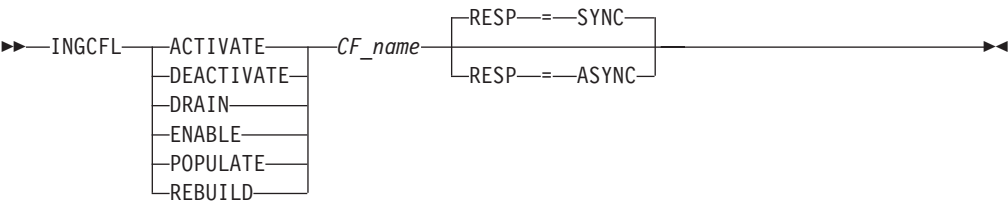
- System-managed rebuild must be supported by all connectors.
 - The structure must be allocated in a coupling facility supporting system-managed duplexing and another coupling facility supporting system-managed duplexing must be defined in its preference list.
 - The CFRM couple data set must support system-managed duplex rebuild and the structure must not have a policy change pending.
 - The structure must be defined in the active CFRM policy when any connection state is not active.
- If you have specified INGCFL with the STR parameter, the **Coupling Facility** field is an input field. To display the structure list of another coupling facility, specify the name of the coupling facility in this field and press ENTER.
- The **Include Condition** field is an input field. By specifying Yes or No in this field you determine whether or not the conditions of the structures are displayed in the **Structure** column.
- The **Permission** field shows your authorization level. There are two possible values, ALL and DISPLAY. DISPLAY signifies that you can only use the display functions. ALL signifies that you can also rebuild and delete structures.
- You can specify an action code before every structure entry. The codes you can enter depend on your authorization level
- The **Structure** column shows the names of the structures.
- The letter P in the **P** column indicates that policy changes are pending for the structure.
- A structure has policy changes pending when it was allocated at the time of a CFRM policy switch, and XES could not bring the switch into effect for that structure. One reason for a pending policy change is that the old and the new policy define the structure differently, for example, with different preference lists.
- The **Condition** column shows the status of the structures. You can switch on and off the display of the conditions through the **Include Condition** field.
- The D field indicates the type of duplexing being possible. The following values are possible:
- | | |
|---|---|
| U | User-managed duplexing |
| S | System-managed duplexing |
| B | User-managed and system-managed duplexing |

INGCFL

Purpose

The INGCFL routine supports line mode for INGCFL. For further information refer to “INGCF” on page 106.

Syntax



INGCFL

Parameters

cfname

Is the name of the CF

ACTIVATE

Activates the coupling facility

DEACTIVATE

Deactivates the coupling facility

DRAIN

Sets the sender paths to OFFLINE after issuing a REBUILD

ENABLE

Sets the sender paths to ONLINE after issuing ACTIVATE

POPULATE

Starts the populate process of the coupling facility after issuing ENABLE

REBUILD

Starts the rebuild process of the coupling facility

RESP

Specifies whether the final result is returned synchronously via return code or asynchronously via message (default: synchronous response)

TARGET

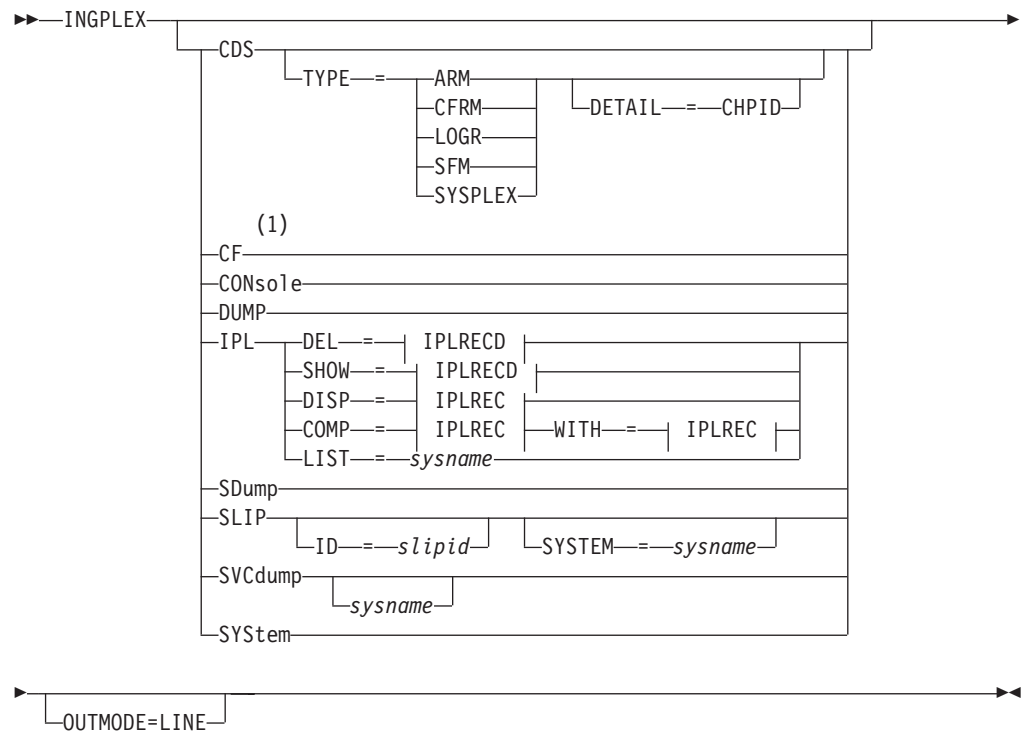
Specifies the system where the command is executed (default: local system)

INGPLEX

Purpose

The INGPLEX command comprises all the sysplex-related functions of msys for Operations. It can be called in full mode and in line mode. If it is called in line mode, only the display functions are available. Therefore, you cannot start an action in msys for Operations if you issue INGPLEX from an MVS console.

Syntax

**IPLREC:**

|—sysname—|/—timestamp—|/—member—|/—suffix—|]

IPLRECD:

|—sysname—|/—timestamp—|

Notes:

- 1 For details see the INGCF command.

Parameters**CDS**

Displays information about CDSs and supports replacement of the current alternate CDS by a new one as well as making the alternate CDS the new primary. For further information about INGPLEX CDS refer to “CDS” on page 128.

TYPE

The type of CDS for which the CDS function is issued. Possible values are ARM, CFRM, LOGR, SFM, and SYSPLEX.

DETAIL

If you specify this parameter with the CDS function, the channel paths for the respective CDS type are displayed.

CF

This is the equivalent of the INGCF command.

CONsole

Displays information about consoles.

DUMP

Shows the DUMP submenu.

IPL

Shows and compares IPL information. It can be issued with the following options:

DEL

Deletes a single IPL record and all its related information. Note that the DEL parameter is supported in line mode only.

SHOW

Shows the details panel of the specified IPL record.

DISP

Shows all, one, or particular PARMLIB members used by the IPL of the specified system and at the specified date and time.

COMP

Compares all, one, or particular PARMLIB members used by the IPL of the specified system and at the specified date and time with those specified in the WITH parameter.

LIST

Shows the IPL summary records of the specified system.

WITH

The COMP parameter compares all, one, or particular PARMLIB members used by the IPL of the specified system, at the specified date and time with those specified in parameter WITH.

sysname

Is the name of the system in the sysplex.

timestamp

Is the IPL date and time. The format is YYYYMMDDhhmm.

member

Is the name of the PARMLIB member without the suffix.

suffix

Is the suffix of the PARMLIB member.

For further information about INGPLEX IPL refer to "IPL" on page 137.

SDUMP

Displays and controls the SDUMP options being set on all systems in the sysplex. For further information about INGPLEX SDUMP refer to "SDUMP" on page 138.

SLIP

Displays and controls all SLIP traps of all systems in the sysplex. Controlling is limited to DISABLE, ENABLE, or REMOVE a SLIP trap. The following parameters are supported:

ID Limits the line mode output and the initial full screen display to the particular SLIP trap ID

slipid

Is the ID of a SLIP trap. It can consist of one to four characters. Wildcards are not supported.

SYSTEM

Limits the line mode output and the initial full screen display to the particular system.

For further information about INGPLEX SLIP refer to “SLIP” on page 144.

SVCDUMP

Allows you to issue a multisystem dump of up to 15 address spaces including data spaces owned by the address spaces, structures used by the address spaces, and XCF group members on the same or on other systems in the sysplex of those groups the address spaces have joined. The following parameter is supported:

sysname

Is the name of the system having joined the XCF group of the NetView the operator is logged on to.

For further information about INGPLEX SVCDUMP refer to “SVCdump” on page 141.

SYStem

Displays information about a member system of the sysplex.

OUTMODE

If you specify OUTMODE=LINE, INGPLEX is called in linemode. In this mode, only the display functions of the command are available.

Example

If you specify INGPLEX without parameters, the following selection panel is displayed:

INGLX000		msys - Command Dialogs	
Domain Id = IPUFH	----- INGPLEX -----	Date = 02/22/02	
Operator Id = NETOP1		Time = 11:10:41	
Sysplex : LOCAL			
Select the desired command:		INGPLEX ...	
1 Display systems (including ETR & signalling paths)		SYStem	
2 Display consoles		CONsole	
3 Control coupling facilities		CF	
4 Control couple data sets		CDS	
6 Display IPL information		IPL	
7 Control dumps		DUMP	
Command ==>			
F1=Help	F2=End	F3=Return	F6=Roll
			F12=Retrieve

Figure 23. INGPLEX selection panel

Specify the number or the function and press ENTER.

INGPLEX CDS

CDS

Purpose

The CDS function displays information about all the couple data sets in the system, including details of the corresponding policies. For every CDS type that is required by the implementation INGPLEX CDS allows the operator to:

- Switch from the primary to the alternate CDS
- Define a new alternate CDS
- Change the active policy (if applicable)

Actions are started by specifying an action code for a selected CDS type on the panel.

Actions

The possible action codes are:

***allocate alternate CDS (A)**

Replaces the current alternate CDS for a selected CDS type by a new one. There are two options how to do this:

- The alternate CDS is allocated automatically by msys for Operations. This automatic allocation requires that spare volumes have been defined in AOFCUST, and that one of these spare volumes is available. For details, see “CDS section – spare volumes for CDS recovery” on page 43.
- Specify the data set that is to be used as the new alternate CDS. If you specify your own data set, observe the following:
 - The data set must exist
 - It must have been formatted with the XCF formatting tool
 - It must be at least as large as the current primary CDS, which means that every value you have passed to the XCF formatting tool (for example, in the case of a sysplex CDS, the maximum number of systems supported) must be equal to or greater than the corresponding value of the primary CDS.

display CHPIDs (C)

Displays information about the channel paths for the selected CDS type.

display CDS information (I)

Displays detail information about the selected CDS type. This comprises the formatting parameters and the policies that are contained in the CDS, if applicable. When the CDSs of the selected type contain policies, the detail information panel provides further actions, namely:

display policy (D)

Displays details about the selected policy.

***start policy (S)**

Makes the selected policy the active policy.

The policy switch must be confirmed before it is executed.

***switch alternate CDS to primary CDS (P)**

Makes the alternate CDS the primary one. Because an alternate CDS is no longer available after the switch, msys for Operations shows a confirmation panel before the action is performed. On the panel, you can specify a new alternate CDS. When CDS recovery is switched on and you do not supply your own alternate CDS, msys for Operations tries to allocate a new alternate CDS automatically. The special requirements for

manual and automatic creation of the new alternate CDS are the same as those for the replacement of the alternate CDS (action code A).

Examples

The following example illustrates the switch from the primary to the alternate CDS.

The following examples start with issuing INGPLEX CDS and pressing F8 on the CDS command dialog to scroll down the CDS list. The following panel is displayed:

```

INGKX300          msys - Command Dialogs          Line 7 of 18
Domain ID = IPSF0 ----- INGPLEX CDS ----- Date = 03/01/01
Operator ID = NETOP1          Sysplex = KEY1PLEX Time = 10:08:10

System...: KEY3              Interval...: 86400      OPNotify: 86400
Maxmsg...: 999999           Cleanup...: 60      Retry...: 255
Classlen: 956               Max CFlevel: 9        COUPLExx: COUPLER1
SMREBLD.: 1                 Max SMlevel: 9

Cmds: A allocate alternate CDS / C display CHPIDs
      D display CDS information / P switch alternate CDS to primary CDS

      Type      MS      Volume  Dev      Couple Dataset Name
      -----
-   CFMR
    PRIMARY...: 16    KEY1SP   260B   SYS1.KEY1.PXESCDs
    ALTERNATE: 16    KEYUSR   261C   SYS1.KEY1.AXESCDs
-   LOGR
    PRIMARY...: 8     KEY1SP   260B   SYS1.KEY1.PLOGCDs
    ALTERNATE: 8     KEYUSR   261C   SYS1.KEY1.ALOGCDs

Command ==>
F1=Help      F2=End      F3=Return      F6=Roll
PF7=Back     PF8=Forward  F9=Refresh    F12=Retrieve
  
```

Figure 24. INGPLEX CDS dialog panel

The panel header contains sysplex-related information about the system on which the INGPLEX command was executed. The details are as follows:

- The **System** field shows the name of the system.
- The **Interval** field shows the system failure detection interval in seconds. This interval is the amount of time XCF lets elapse without a status update before assuming that the system failed.
- The **OPNotify** field shows the number of seconds that XCF waits before notifying the operator of a potential system problem.
- The **Maxmsg** field shows the default value for the maximum amount of kilobytes of message buffer space. This default value is used when MAXMSG is not specified on SETXCF START commands.
- The **Cleanup** field shows the number of seconds that XCF waits for cleanup of members.
- The **Retry** field shows the default value for the retry limit. This value is used when the RETRY keyword is not specified on SETXCF START commands.
- The **Classlen** field shows the default length (in bytes) of messages allowed for a transport class. This value is used when CLASSLEN is not specified on the SETXCF START CLASSDEF command.
- The **Max CFlevel** field shows the maximum CFLEVEL supported by this system. This system can connect to a coupling facility with a higher CFLEVEL than the value of **Max CFlevel** but would not be enabled to use any functions supported by the higher level coupling facility.

INGPLEX CDS

- The **COUPLExx** field shows the COUPLExx Parmlib member used for system IPL.
- The **SMRBLD** field shows whether (value 1) or not (value 0) system-managed rebuild has been activated in the CFRM couple dat set.
- The **Max SMlevel** field shows the maximum system-managed process level supported by this system.

The main part of the screen shows information about the primary and alternate CDSs for every CDS type. Press F8 to scroll and display further entries. The **MS** column shows the maximum number of systems that are supported by the CDS.

Making an alternate CDS the primary CDS: In this example, the alternate LOGR couple data set is made the new primary CDS. A new alternate CDS is automatically generated.

To switch the LOGR couple data set, enter P before LOGR on the panel displayed in Figure 24 on page 129, and press ENTER. INGPLEX CDS displays the following confirmation panel:

```
INGKX30A          msys - Command Dialogs
Domain ID  = IPSFO  ----- INGPLEX CDS -----      Date = 03/01/01
Operator Id = NETOP1      Sysplex = KEY1PLEX          Time = 10:08:13

                        SETXCF PSWITCH Confirmation

You are going to remove the LOGR   primary couple data set.
The alternate couple data set  SYS1.KEY1.ALOGCDS
becomes the primary as soon as you proceed with the GO function key.
Immediately after the switch, automation will try to allocate a new alternate
couple data set on one of the spare volumes defined during the customization.
If you want the automation to allocate your own alternate couple data set
complete the necessary information below.

Your alternate couple dataset...

Name   ==>

Volume ==>

Command ==>
        F2=End      F3=Return      F10=Go      F11=Cancel  F6=Roll
                                   F12=Retrieve
```

Figure 25. Confirmation panel for switching from the current primary CDS to the alternate CDS

Use this panel to determine how a new alternate CDS is to be created after the switch. You can either specify your own new alternate CDS or let msys for Operations create it for you. When you specify the new alternate CDS yourself, the data set must exist and must have been formatted with the XCF formatting tool. Automatic creation requires that spare volumes have been defined for LOGR couple data sets in the AOFCUST customization file.

Pressing F10 causes msys for Operations to generate the new alternate CDS. After returning to the CDS command dialog, refreshing the panel, and scrolling down with F8, the panel looks as follows:

```

INGKX300      msys - Command Dialogs      Line 7 of 18
Domain ID = IPSFO ----- INGPLEX CDS ----- Date = 03/01/01
Operator ID = NETOP1      Sysplex = KEY1PLEX      Time = 10:08:25

System..: KEY3      Interval...: 86400      OPNotify: 86400
Maxmsg...: 999999      Cleanup....: 60      Retry...: 255
Classlen: 956      Max CFlevel: 9      COUPLExx: COUPLER1
SMREBLD.: 1      Max SMlevel: 9

Cmds: A allocate alternate CDS / C display CHPIDs
      D display CDS information / P switch alternate CDS to primary CDS

      Type      MS      Volume      Dev      Couple Dataset Name
      -----
-   CFRM
      PRIMARY...: 16      KEY1SP      260B      SYS1.KEY1.PXESDCS
      ALTERNATE: 16      KEYUSR      261C      SYS1.KEY1.AXESDCS
-   LOGR
      PRIMARY...: 8      KEYUSR      261C      SYS1.KEY1.ALOGCDS
      ALTERNATE: 8      AOCUSR      262B      AOC.CDS.TEST.LOGR.CDS02

Command ==>
F1=Help      F2=End      F3=Return      F6=Roll
PF7=Back      PF8=Forward      F9=Refresh      F12=Retrieve

```

Figure 26. INGPLEX CDS dialog panel after the switch

The previous alternate LOGR CDS has become the primary, and there is a new alternate, which was created by msys for Operations according to the specifications in the CDS section of AOFCUST. See “CDS section – spare volumes for CDS recovery” on page 43 for details.

Switching the CFRM policy: In this example, the active CFRM policy is switched.

Enter D before CFRM on the panel displayed in Figure 24 on page 129, and press ENTER. The following panel is displayed:

```

INGKX311      msys - Command Dialogs      Line 1 of 5
Domain ID = IPSFO ----- INGPLEX CDS ----- Date = 03/01/01
Operator ID = NETOP1      Sysplex = KEY1PLEX      Time = 10:13:13
                        CFRM Couple Data Set Information

Data Set Information
Volume Device FORMAT TOD      Data Set Name
-----
KEY1SP 260B 08/29/2000 08:51:30 SYS1.KEY1.PXESDCS
KEYUSR 261C 08/29/2000 08:47:42 SYS1.KEY1.AXESDCS
Control Card Information
MS POLICY CF STR CONNECT SMREBLD SMDUPLEX
--
16      8 4 64      16      1      0
Policy Information
Cmds: D display policy / S start policy
      Name      CF Str Date      Time      Userid
      -----
-   BZOEPOl ACTIVE 2 19 02/10/2001 10:05:47 BZOE
-   HIRPOL      2 19 02/19/2001 19:45:57 HIR
-   HIRPOL1     1 8 08/25/2000 09:20:04 HIR

Command ==>
F1=Help      F2=End      F3=Return      F6=Roll
PF8=Forward      F9=Refresh      F12=Retrieve

```

Figure 27. CFRM couple data set information panel before policy switch

The panel shows information about the names and locations of the CDSs. The panel also shows the parameters that were used by the formatting tool of XCF for the allocation of the CDS. The **POLICY** field, for example, displays the maximum

INGPLEX CDS

number of policies the CDS can contain. Furthermore, the panel shows information about the policies in the CDS, for example, how many coupling facilities and structures are defined in every policy, and which policy is currently active.

To switch to the HIRPOL policy, enter S before this policy and press ENTER. INGPLEX CDS displays the following confirmation panel:

```

INGKX30C          msys - Command Dialogs
Domain ID   = IPSFO  ----- INGPLEX CDS -----
Operator Id = NETOP1      Sysplex = KEY1PLEX          Date = 03/01/01
                                                Time = 10:13:17

                SETXCF START Confirmation

You are going to start a new CFRM CDS policy named HIRPOL .

The current policy

                BZOEPOL

will be stopped as soon as you proceed with the GO function key,

Command ==>

                F2=End          F3=Return          F10=Go          F6=Roll
                                                F11=Cancel    F12=Retrieve

```

Figure 28. Confirmation panel for policy switch

Displaying the channel paths for a CDS type: In this example, the channel paths for the CFRM couple data sets are displayed.

Enter C before CFRM on the panel displayed in Figure 24 on page 129, and press ENTER. The following panel is displayed:

```

INGKX318          msys - Command Dialogs          Line 1 of 4
Domain ID   = IPSFO  ----- INGPLEX CDS -----   Date = 03/02/01
Operator ID = NETOP1      Sysplex = KEY1PLEX          Time = 08:05:46

                CFRM Channel Path Information

System  T  DEVN  CHPIDs                      SSID
-----  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
KEY1    P  260A  E4=+ E5=+ E2=+ E3=+          2600
        A  2610  E4=+ E5=+ E2=+ E3=+          2600
KEY2    P  260A  E4=+ E5=+ E2=+ E3=+          2600
        A  2610  E4=+ E5=+ E2=+ E3=+          2600
KEY3    P  260A  E4=+ E5=+ E2=+ E3=+          2600
        A  2610  E4=+ E5=+ E2=+ E3=+          2600
KEY4    P  260A  13=+ 22=+ 30=+ 94=+         2600
        A  2610  13=+ 22=+ 30=+ 94=+         2600

Command ==>
F1=Help      F2=End          F3=Return          F6=Roll
F9=Refresh                    F12=Retrieve

```

Figure 29. Channel path information for CFRM couple data sets

- The System column shows the name of the sysplex members.
- The T column (for 'type') indicates whether the CDS is the primary (value 'P') or alternate (value 'A').
- The DEVN displays the number of the device on which the CDS resides.
- The CHIDs column shows the status of the paths to the devices in the format *chpid=status_code*. The codes are those of the operating system. They have the following meaning:

+	The path is logically and physically available and I/O on the path was successful.
*	The path is physically, but not logically available. The subchannel's logical path indicator is off but I/O to the path is successful. You can use the command <code>VARY PATH (ddd,nn),ONLINE</code> to make channel path <i>nn</i> logically available to device <i>ddd</i> .
-	The path is neither logically nor physically available. The subchannel's logical and physical indicators are both off for this channel path. You can use the command <code>CONFIG CHP(nn),ONLINE</code> to make the channel path logically available to all devices connected to the channel.
&	The device is reserved to another path. This indicator applies to devices with the dynamic pathing selection feature.
<	The path is installed but not physically available. The start subchannel request received a condition code of 3.
>	The device microcode has detected an error and will not allow I/O to complete on the path.
B	The path is unable to communicate. The device indicates that a busy or reserve condition exists on the path.
C	A controller error occurred while accessing the device.
D	A device error occurred while accessing the device.
I	Intervention is required; the device is not ready.
R	The path is available and the device is reserved to this path/group. This only applies to devices with the dynamic pathing feature.
T	A time out has occurred; there is no response from the device. The cause of the time out is undetermined and this condition is transient.
U	A storage control unit or storage director error occurred while accessing the device.
X	Unable to determine the failing unit.
- The SSID field displays the storage subsystem to which the device belongs.

SYStem

Purpose

The SYSTEM function displays the target sysplex name, its GRS mode and its member systems.

Example

```

AOFKX100      msys - Command Dialogs      Line 1 of 4
Domain ID = IPSFP ----- INGPLEX SYSTEM ----- Date = 03/05/01
Operator ID = NETOP1                               Time = 09:44:37

Sysplex . . . . . : KEY1PLEX
GRS Mode . . . . . : STAR

Display more info: C CPU E ETR I IPL O IOS S STOR/ESTOR
Signalling Path : D device T structure

      Monitor
Cmd  System  Status  Timestamp  INTERVAL  Action  SSUM  TIME  WEIGHT
-----
-    KEY1    ACTIVE  09:44:34  86400     ISOLATE  50    50    50
-    KEY2    ACTIVE  09:44:35  86400     ISOLATE  50    15    15
-    KEY3    ACTIVE  09:44:34  86400     ISOLATE  50    15    15
-    KEY4    ACTIVE  09:44:36  86400     ISOLATE  50    15    15

Command ==>
F1=Help      F2=End      F3=Return      F6=Roll
              F9=Refresh      F12=Retrieve

```

Figure 30. INGPLEX SYSTEM dialog panel

The following command codes are available:

- C** Displays the online or offline status of one or more processors and any vector facilities, or ICRFs attached to those processors
 - E** Displays the timer synchronization mode and ETR ports
 - I** Displays IPL information
 - O** Displays IOS-related configuration information
 - S** Displays the number of megabytes of central and expanded storage assigned and available to the system
 - D** Displays the device number of one or more in-/outbound signalling paths that XCF can use and information about in-/outbound XCF signalling paths to this system
 - T** Displays detailed signalling path information for all coupling facility structures
 - The Sysplex field shows the name of the sysplex.
 - The GRS Mode field shows the GRS mode of the target system. The mode can be either STAR or RING.
 - The CMD column allows you to specify command codes. To use one of the command codes shown, type the appropriate letter next to the resource name, and press ENTER.
 - The System column shows the name of the system.
 - The Status column shows the status of the system.
 - The Monitor Timestamp column shows the last time stamp recorded for status monitoring on this system.
 - The INTERVAL column shows the system failure detection interval in seconds. This interval is the time XCF allows to elapse without a status update before assuming that the system failed.
- The last three columns contain configuration data of the SFM policy (if applicable).
- The SSUM Action field shows the SSUM action. It can be one of the following:

- ISOLATE
- DEACTIVATE
- RESET
- PROMPT
- N/A
- The SSUM TIME field shows the SSUM interval as specified in the current SFM policy.
- The SSUM WEIGHT field shows the SSUM weight specified in the current SFM policy. This value is used in sysplex reconfigurations after a signalling connectivity failure.

CONsole

Purpose

The CONSOLE function displays the following information for the sysplex:

- The name of the master console
- WTO & WTOR buffer utilization
- Number of queued messages (replies) of various types
- Awaiting mounts
- Operator requests and list of consoles (name, status, authority, number of WTOR buffers, UD, device, system, ALTGRP, MSCOPE)

Example

```

INGLX400          msys - Command Dialogs          Line 1 of 6
Domain Id = IPSFP  ----- INGPLEX CONSOLE ----- Date = 04/12/01
Operator Id = NETOP1                               Time = 10:36:26

Sysplex . . . . . : KEY1PLEX          Master Console . . . : --none--
Message Buffer Usage : 14 / 9999        Reply Buffer Usage . : 14 / 99
Awaiting Replies . . : 14              Eventual Action . . : 0
Immediate Action . . : 0              Awaiting Mounts . . : 0
Critical Action . . : 0              Operator Requests . : 0
-----

Cmds: D Details / R Requests

  Console  Status  AUTH  NBUF  UD  Device  System  ALTGRP  MSCOPE
  -----
MASTER    INACTIVE MASTER n/a   Y   -none-  --none-- --none-- *ALL
MASTER1   INACTIVE ALL   n/a   N   -none-  --none-- --none-- *ALL
03         INACTIVE MASTER n/a   N   -none-  --none-- --none-- *ALL
04         INACTIVE MASTER n/a   N   -none-  --none-- --none-- *ALL
05         INACTIVE MASTER n/a   N   -none-  --none-- --none-- *ALL
06         INACTIVE MASTER n/a   N   -none-  --none-- --none-- *ALL

Command ==>
F1=Help      F2=End      F3=Return    F6=Roll
              F9=Refresh   F12=Retrieve

```

Figure 31. INGPLEX CONS dialog panel

The following command codes are available:

D Displays details for the console

R Displays the actual requests for the console

- The **Sysplex** field shows the name of the sysplex.
- The **Message Buffer Usage** field shows the limit of the number of WTO message buffers allowed outstanding.

INGPLEX CONSOLE

- The **Awaiting Replies** field shows a decimal number representing the number of messages awaiting replies.
- The **Immediate Action** field shows a decimal number representing the number of outstanding immediate action messages (with descriptor codes 1 or 2). If the number is greater than 99999, asterisks appear in this field.
- The **Critical Action** field shows a decimal number representing the number of outstanding critical eventual action messages (with descriptor code 11). If the number is greater than 99999, asterisks appear in this field.
- The **Master Console** field shows the name of the master console.
- The **Reply Buffer Usage** field shows the limit of the number of WTOR message buffers allowed outstanding. The maximum value of yyyy is specified by the RMAX parameter in the CONSOLxx parmlib member.
- The **Eventual Action** field shows a decimal number representing the number of outstanding eventual action messages (with descriptor code 3). If the number is greater than 99999, asterisks appear in this field.
- The **Awaiting Mounts** field shows a decimal number representing the number of outstanding mount requests.
- The **Operator Requests** field shows a decimal number representing the number of outstanding requests for operator intervention.
- The **CMD** column lets you specify the command codes shown on the panel. Type the appropriate letter next to the resource name, and press ENTER.
- The **Console** column shows the name of the console as specified in the CONSOLxx parmlib member.
- The **Status** field shows the status of the console. The following values can occur:

HARDCOPY

Hardcopy log. This condition is only indicated if the console is active on the system where the command processes.

ACTIVE

Active console

ACTIVE-P

In the process of becoming an active console. This condition is only indicated if the console is active on the system where the command is processing.

MASTER

Master console

INACTIVE

Inactive console

INACT-P

In the process of becoming a non-active console. This condition is only indicated if the console is active on the system where the command is processing.

PROB-DET

The active system console is in the problem determination mode. PD is indicated only for the system console.

SUBSYS

Subsystem-allocatable console

- The **AUTH** column shows which commands may be entered from this console. The following values can occur:

ALL Any INFO SYS, IQ or CONS command may be entered from this console.

CONS INFO commands and any commands from the console command group may be entered from this console.

INFO Any command from the informational command group may be entered from this console.

IO INFO commands and any commands from the I/O Control command group may be entered from this console.

MASTER

The specified console is authorized to enter any operator command.

NONE

This console has no command authority.

SYS INFO commands and any commands from the system control command group may be entered from this console.

- The **NBUF** column shows the number of WTO message buffers currently queued to this console. If nnnn is greater than 9999, asterisks (****) appear in this field.
- The **UD** column shows whether this console is receiving messages with the UD attribute.
- The **Device** column shows the name of the console as specified in the CONSOLxx parmlib member. This console can be a subsystem allocatable console.
- The **System** column shows the device number of the active console.
- The **ALTGRP** column shows the alternate group defined for this console.
- The **MSCOPE** column lists the name of the system or systems from which this console is receiving unsolicited messages. Note that these systems might be different from the system where this console is physically attached.

IPL

Purpose

With the INGPLEX IPL command you can view and compare the IPL information of the operating system. If a system does not behave after IPL as expected, the IPL recording function enables you to identify parameters that were changed, for example, since the last IPL. The recording function enables you to compare different IPL scenarios. INGPLEX IPL is a tool that helps to identify and resolve the cause of startup problems. The following information can be displayed:

- The selected system (or blank)
- The name of the sysplex
- The maximum number of IPLs that are stored for each system
- An indicator showing whether comments in PARMLIB members are ignored when collecting information

Example

```

INGLX200      msys - Command Dialogs      Line 1 of 6
Domain ID = IPSFM ----- INGPLEX IPL ----- Date = 02/22/02
Operator ID = NETOP1                               Time = 17:59:27

System . . . . . ==>      Max. number of IPL records/system : 10
Sysplex . . . . . ==> KEY1PLEX  Suppression of PARMLIB comments . : N
-----
Cmds: C compare record / D display details / E erase record

  System  IPL Timestamp      Dev  Volume  OpSys  Release  FMID
  -----
  KEYA    2002-02-22 13:52    770E 120204  z/OS   SP7.0.2  HBB7705
  KEYA    2002-02-09 09:28    770E 120204  z/OS   SP7.0.2  HBB7705
  KEYA    2002-02-08 15:28    770E 120204  z/OS   SP7.0.2  HBB7705
  KEYA    2001-12-10 14:31    0707 120147  z/OS   SP7.0.2  HBB7705
  KEYB    2002-02-22 13:59    770E 120204  z/OS   SP7.0.2  HBB7705
  KEYB    2002-02-14 16:24    770E 120204  z/OS   SP7.0.2  HBB7705
  KEYB    2002-02-11 18:46    770E 120204  z/OS   SP7.0.2  HBB7705
  KEYB    2002-02-11 15:36    770E 120204  z/OS   SP7.0.2  HBB7705
  KEYB    2002-02-11 14:22    770E 120204  z/OS   SP7.0.2  HBB7705

Command ==>
F1=Help      F2=End      F3=Return      F6=Roll
              F8=Forward  F9=Refresh    F10=Previous  F11=Next      F12=Retrieve

```

Figure 32. INGPLEX IPL main panel

Use F10 and F11 to scroll through all available columns. SORT by column numbers is supported as well as the FIND and RFind command to locate information on the panel. You can also limit the display to a particular system by specifying the system name in the appropriate entry field.

The following command codes are available:

- C** Compares the complete IPL information with another IPL record. A second panel will be displayed where you can select the second record.
- D** Displays detailed information about this IPL record.
- E** Erases the IPL information records. This action must be confirmed.
 - The Sysplex field shows the name of the sysplex.
 - The System field shows the name of the system in the sysplex.
 - The IPL Timestamp field shows the date and time of the IPL. The format is YYYY-MM-DD HH:MM converted to local timezone.
 - The Dev field shows the IPL device number.
 - The Volume field shows the volume serial of the IPL device.
 - The OpSys field shows the name of the operating system, for example, z/OS or OS/390.
 - The Release field shows the release level of the operating system.
 - The FMID field shows the FMID of the operating system.

For further information about the panel fields refer to the online help.

SDUMP**Purpose**

The INGPLEX SDUMP command lets you control the default dump options sysplex-wide.

Example

The dump functions can be invoked directly by specifying the commands, or from the dump panel of the INGPLEX command selecting the appropriate command. In addition, you can invoke the dump submenu from the main panel of the INGPLEX command selecting command 7. The following panel is displayed:

```

INGLX250      msys - Command Dialogs      Line 1 of 12
Domain Id = IPSFP ----- INGPLEX ----- Date = 02/26/02
Operator Id = NETOP1                      Time = 16:30:36

Sysplex . . . . . : KEY1PLEX

Select the desired command:                INGPLEX ...

  1 Control default SDUMP options          SDUMP
  2 Issue SVC dumps                       SVC DUMP
  3 Control SLIP trap settings             SLIP

Command ==>
F1=Help      F2=End      F3=Return      F6=Roll
F12=Retrieve

```

Figure 33. INGPLEX dump options panel

If you select option 1, the following panel is displayed:

```

INGLX251      msys - Command Dialogs      Line 1 of 12
Domain Id = IPSFP ----- INGPLEX SDUMP ----- Date = 02/26/02
Operator Id = NETOP1                      Time = 15:44:58

Sysplex . . . . . ==> KEY1PLEX      Permission . . . . . : ALL
-----
Cmds: C change

System  Dump options
-----
KEY1    Q=      Type=      Buffers= 0K MaxSpace= 500M MsgTime=99999
                        LSQA
                        TRT
KEY2    Q=      Type=      Buffers= 0K MaxSpace= 500M MsgTime=99999
                        LSQA
                        TRT
KEY3    Q=      Type=      Buffers= 0K MaxSpace= 500M MsgTime=99999
                        LSQA
                        TRT
KEY4    Q=      Type=      Buffers= 0K MaxSpace= 500M MsgTime=99999

Command ==>
F1=Help      F2=End      F3=Return      F6=Roll
F8=Forward   F9=Refresh   F12=Retrieve

```

Figure 34. INGPLEX SDUMP panel

The following command code is available:

C change

Invokes the modification panel by providing the options of the selected system as input

INGPLEX SDUMP

- The Sysplex field shows the name of the sysplex.
- The System field shows the name of the system in the sysplex.
- The Permission field shows your authorization level.
- The Dump options field shows the default SDUMP options of all systems in the sysplex. For each system the following details are displayed:

Q= Shows whether or not SDUMP quiesces the system while dumping the contents of the SQA or CSA.

TYPE= Causes SVC dump to dump the cross memory address spaces that the caller has when SVC dump gets control (XMEM) or when the error causing the dump occurs (XMEME).

BUFFERS= Shows the reserved storage exclusively used by SVC dump. This storage can be used while capturing the contents of the common area storage.

MaxSpace Shows the maximum amount of virtual storage that SVC dump can use to capture volatile virtual storage data, summary dump data, and component-specific data before writing the dump to DASD.

MsgTime Shows for which amount of time (mm) the message IEA793A is shown at the console. When the system deletes the message, it also deletes the captured dump.

The FIND and RFIND commands are supported. If you specify command code C, the following panel is displayed:

```

INGLX252          msys - Command Dialogs          Line 1 of 12
Domain Id = IPSFP  ----- INGPLEX SDUMP -----   Date = 02/26/02
Operator Id = NETOP1                                Time = 16:18:08

System . . . . . : KEY1
Sysplex . . . . . : KEY1PLEX          Recommended options are underlined.

NODUMP ... ==> N      (all other options below are ignored)

ALLNUC ... ==>      ALLPSA(*) ==>      COUPLE ... ==>      CSA ..... ==>
GRSQ .... ==>      LPA ..... ==>      LSQA ..... ==> Y      NUC ..... ==>
PSA ..... ==>      RGN ..... ==>      SERVERS .. ==>      SQA(*) ... ==>
SUMSUMP(*) ==>      SWA ..... ==>      TRT ..... ==> Y      WLM ..... ==>
XESDATA .. ==>      (*) = The NOxxx option is generated when not selected.

Q(uiesce) ==>      (YES / NO)
Type ..... ==>      (XMEM / XMEME)
Buffers .. ==> 0K    (nnnnK / nnnM)
MaxSpace . ==> 500   (MB)
MsgTime .. ==> 99999 (minutes)

Command ==>
F1=Help      F2=End      F3=Return      F4=Set SYS      F5=Undo all      F6=Roll
F10=Set SYSS F11=Set SYSP F12=Retrieve

```

Figure 35. INGPLEX SDUMP modification panel

The modification panel allows to modify all SDUMP options. Furthermore, you can delete SDUMP options. After entering your changes you can set the new options for:

- The selected system
- All systems in the sysplex

- Selected systems in the sysplex

To set the options press the appropriate F-key. If you want to modify selected systems in the sysplex, you are prompted for the systems on which the SDUMP options are being changed. To reset the options to the state when the modification panel was invoked press F5 Undo all.

Note: The user must be authorized to change any SDUMP option. The authorization can be any of those which are used for controlling coupling facilities and couple data sets. For further information refer to .

For further information about the panel fields refer to the online help.

SVCdump

Purpose

The INGPLEX SVCDUMP function allows you to issue a multisystem dump of up to 15 address spaces of a single system including their data spaces and structures.

Example

```

INGLX26S          msys - Command Dialogs          Line 1 of 6
Domain Id  = IPSFP  ----- INGPLEX SVCDUMP -----   Date = 02/06/02
Operator Id = NETOP1                                Time = 17:05:17

The following systems of sysplex KEY1PLEX are registered to the automation.
Use any non-blank character to select one system and then press ENTER.

Sel  System
---  -----
-    KEY2
-    KEY3
-    KEY4

Command ==>
          F1=Help    F2=End    F3=Return    F6=Roll
                               F12=Retrieve
  
```

Figure 36. INGPLEX SVCDUMP target system selection panel

- The Sel field lets you select a system from which a multisystem dump is issued.
- The System field shows the name of the system having joined the same XCF group the operator is logged on to.

For further information about the panel fields refer to the online help. After selecting a system and pressing ENTER, the following panel is displayed:

INGPLEX SVCdump

```

INGLX260          msys - Command Dialogs          Line 38 of 63
Domain Id  = IPXFG  ----- INGPLEX SVCDUMP -----      Date = 03/11/02
Operator Id = NETOP1                                     Time = 12:26:26

System . . . . . ==> KEYA
Sysplex . . . . . ==> KEYAPLEX
-----
Cmds: D/S de-/select job names for the SVC dump (up to 15 can be specified)

  Jobname      ASID  WorkUnitID  Userid
  -----
- PCAUTH       0002
- PORTMAP      0042  STC06027  PORTMAP
- RACF         0023  STC05974  STCUSER
- RASP         0003
- RESOLVER     001D  STC05973  TCPIP
- RMF          0221  STC06506  STCUSER
- RRS          0046
- SGA1DBM 1    0044  STC06035  SGAUSER
- SGA1DIST     0041  STC06038  SGAUSER
- SGA1IRLM     0043  STC06032  SGAUSER
- SGA1MSTR     001C  STC06028  SGAUSER
- SMF          0015
- SMS          0008
- SYSBMAS      0009
- TCPIP        001A  STC05986  TCPIP
- TNF          0024
- TRACE        0004
- TSO          003B  STC05983  STCUSER
- VLF          0019
- VMCF         0025
- VTAM         001E  STC05982  STCUSER          selected
- WATS         0217  TSU06587  -                selected
- WLM          000B
- XCFAS        0006

Command ==>
F1=Help      F2=End      F3=Return      F5=NextPnl      F6=Roll
              F8=Forward  F9=Refresh

```

Figure 37. INGPLEX SVCDUMP address space selection panel

If you select VTAM address space and WATS address space, which is a user, press ENTER, then F5, the following panel is displayed:

```

INGLX261          msys - Command Dialogs          Line 1 of 15
Domain Id  = IPXFG  ----- INGPLEX SVCDUMP ----- Date = 03/11/02
Operator Id = NETOP1                               Time = 12:34:04

System . . . . . : KEYA
Sysplex . . . . . : KEYAPLEX
-----
Cmds: D/S de-/select the areas to be dumped (max. 113 structures)
      A selection of the job name includes all related areas.

  Jobname  ASID  T Data Space/XCF Group Member/Structure
  -----
  _VTAM    001E
  -          D ISTNMPDS
  -          D ISTNMSDS
  -          D IST08648
  -          D IST56892
  -          D IST69B2F
  -          D IST9BD36
  -          D IST90C95
  -          D 00012IXL
  -          D 00013IXL
  -          L ISTGENERIC
  -          M ISTCFS01.KEYB.VTAM.IPXVH__DEIBMIPS  selected
  -          M ISTXCF.KEYB.VTAM.IPXVH__DEIBMIPS  selected
  -          M IXCLO008.KEYB.VTAM.M28            selected
  - WATS    0217
Command ==>
F1=Help      F2=End      F3=Return  F4=PrevPnl  F5=NextPnl  F6=Roll
              F8=Forward  F9=Refresh

```

Figure 38. INGPLEX SVCDUMP address space detail panel

Address space VTAM has many dataspace (D), one list structure (L) and some XCF group members (M). TSO user WATS has nothing.

The following command codes are supported:

- D** Deselects the previous selection
- S** Selects a local address space, data space, structure, or XCF group member address space for the SVC dump.

If you press F5, the dump option selection panel is displayed:

INGPLEX SVCdump

```
INGLX262          msys - Command Dialogs
Domain Id   = IPSFP ----- INGPlex SVC -----
Operator Id = NETOP1                                     Date = 02/26/02
                                                         Time = 18:02:56

System . . . . . : KEY3
Sysplex . . . . . : KEY1PLEX

Title .... ==>
                ==>

SDATA Dump Options (recommended options are underlined)
ALLNUC ... ==>    ALLPSA(*) ==> Y    COUPLE ... ==>    CSA ..... ==> Y
GRSQ ..... ==> Y  LPA ..... ==>    LSQA ..... ==>    NUC ..... ==> Y
PSA ..... ==>    RGN ..... ==> Y    SERVERS .. ==>    SQA(*) ... ==> Y
SUMSUMP(*) ==> Y  SWA ..... ==>    TRT ..... ==> Y    WLM ..... ==>
XESDATA .. ==>    (*) = The NOxxx option is used when not selected.

Structure Dump Options (SUMMARY and ADJUNCT/ENTRYDATA are mutually exclusive)
COCLASS .. ==>    EMCONTROLS ==>    LISTNUM .. ==>    STGCLASS . ==>
ADJUNCT .. ==>    ENTRYDATA ==>    SUMMARY .. ==>

Command ==>
F1=Help      F2=End      F3=Return   F4=PrevPnl  F5=Dump      F6=Roll
F12=Retrieve
```

Figure 39. INGPlex SVCDUMP dump option panel

The panel shows the default dump options being set on invocation. After specifying the dump title, press F5 to issue the dump. When the dump is taken, the function returns to the address space selection panel with all selections cleared. The SORT, FIND and RFIND commands are supported for selection panels only. For further information about the panel fields refer to the online help.

SLIP

Purpose

With the INGPlex SLIP command you can display serviceability level indication processing (SLIP) traps being set at all systems in the sysplex. With INGPlex SLIP you can view, enable, disable, and delete the SLIP trap defined in the sysplex.

Example

```
INGLX270      msys - Command Dialogs      Line 1 of 6
Domain Id = IPSFP      ----- INGPLEX SLIP -----      Date = 02/26/02
Operator Id = NETOP1      Time = 18:20:21

System . . . . . ==>      (leave blank for all systems)
Slip Trap Id . . ==>      (leave blank for all ids)
Sysplex . . . . . ==> KEY1PLEX      Permission . . . . : ALL
-----
Cmds: +/- display/hide settings D disable E enable R remove

  System   Id    Status   Settings
-----
- KEY1     XB37  ENABLED
- KEY1     XD37  ENABLED
- KEY1     XE37  ENABLED
- KEY1     X0E7  ENABLED
- KEY1     X0F3  ENABLED
- KEY1     X013  ENABLED
- KEY1     X028  ENABLED
- KEY1     X13E  ENABLED

Command ==>
F1=Help      F2=End      F3=Return      F6=Roll
              F8=Forward  F9=Refresh     F12=Retrieve
```

Figure 40. INGPLEX SLIP main panel

The following command codes are available:

- + Shows the settings of the SLIP trap.
- Hides the settings of the SLIP trap.
- D Disables the SLIP trap.
- E Enables the SLIP trap.
- R Deletes the SLIP trap.

The SORT, FIND and RFIND commands are supported.

Note: The user must be authorized to enable, disable, and delete a SLIP trap. The authorization can be any of those which are used for controlling coupling facilities and couple data sets.

For information about the panel fields refer to the online help.

INGPLEX SLIP

Chapter 15. Debugging and supporting commands

ACF

Purpose

The ACF command is used to load automation control file data. You can use ACF to refresh data of a particular system if it does not affect automation manager configuration data.

The ACF COLD command is used to activate configuration changes you have made in the msys for Operations configuration file AOFCUST.

The ACF STATUS command displays internal information about generating the ACF with the msys for Operations configuration file AOFCUST.

Messages

The following is a list of messages that are issued during the operation of ACF.

For the load function (COLD):

```
AOF042I MEMBER ACFZ999 NOT FOUND
AOF100I 16:05:09 : 'ACF REFRESH' COMMAND ISSUED
AOF618I NO VALID ACF FOUND FOR sysname - detail description
AOF782I AUTOMATION CONTROL FILE PROCESSING COMPLETED
```

For the status function (STATUS,CACHE):

```
AOF005I MEMBER ACFZ999 CURRENTLY BEING SAVED IN THE CACHE
AOF006I BUILT      BY opid      ON 02/22/00 AT 18:05:17
AOF006I SAVED      BY opid      ON 03/09/00 AT 16:42:51
AOF006I CONFIGURATION TOKEN = 200002221801085A0345229673
AOF006I CONFIGURATION DATASET = AOC.GDGS.ACF.G0007V00
AOF002I END OF MULTI-LINE MESSAGE GROUP
```

AOCTRACE

Purpose

The AOCTRACE command turns the debugging feature on or off. It works with both a global debugging feature, and two automation procedure (REXX EXEC) specific ones.

Examples

If you type aoctrace on you see a message indicating that the debugging facility is enabled. While the debugging facility is enabled, message AOF700I is written to the netlog for each procedure being processed.

AOCTRACE

```
AOFKAAND          msys - Command Dialogs
Domain ID = IPSNL ----- AOCTRACE ----- Date = 04/27/01
Operator ID = NETOP1                               Time = 13:43:59

Current global mode is ON                        Trace Settings

Current clist is ingrx000                        A All
debug is                                         R Results
trace is r                                     I Intermediates
                                              C Commands
                                              E Errors
                                              F Failures
Select option:                                  L Labels
1 Turn global execution trace OFF              O Off
2 Turn specific clist debug                    N Normal
3 Show current clist debug settings            _ Default
4 Display clists being traced

Specify Subroutines to be traced: (* for all)
*

Command ==> 2
PF1=Help PF2=End PF3=Return PF6=Roll PF12=Retrieve
```

Figure 41. AOCTRACE Command Dialog Panel 1

If you press Enter, a panel similar to the following is displayed:

```
AOFKAAND          msys - Command Dialogs
Domain ID = IPSNL ----- AOCTRACE ----- Date = 04/27/01
Operator ID = NETOP1                               Time = 13:44:16

Current global mode is ON                        Trace Settings

Current clist is INGRX000                        A All
debug is ON                                       R Results
trace is R                                     I Intermediates
                                              C Commands
                                              E Errors
                                              F Failures
Select option:                                  L Labels
1 Turn global execution trace OFF              O Off
2 Turn specific clist debug OFF                N Normal
3 Show current clist debug settings            _ Default
4 Display clists being traced

Specify Subroutines to be traced: (* for all)
*

AOF095I AOCTRACE INGRX000,ON,R FUNCTION SUCCESSFULLY COMPLETED
Command ==>
PF1=Help PF2=End PF3=Return PF6=Roll PF12=Retrieve
```

Figure 42. AOCTRACE Command Dialog Panel 2

To switch the trace for member `ingrx000` on, type the following:

```
aoctrace ingrx000,on,r
```

To switch the trace for member `ingrx000` off, type the following:

```
aoctrace ingrx000,off,r
```

Messages

The following messages are issued:

```
AOF302I 13:44:52 : REQUEST AOFRADBG INGRX000 ON R BY NETOP1 IS COMPLETED FOR NETOP1
```

```
AOF302I 13:45:18 : REQUEST AOFRADBG INGRX000 OFF BY NETOP1 IS COMPLETED FOR NETOP1
```


DISPACF

Purpose

The DISPACF command displays resource information and automation policy settings for a specific entry or entry-type pair in the automation control file.

Examples

If you type `dispacf subsystem`, a panel similar to the following is displayed:

```

AOFK3D0X          msys - Command Response      Line 1 of 10
Domain ID  = IPSN7  ----- DISPACF -----   Date = 06/13/01
Operator ID = NETOP1                                Time = 17:41:10

Command = ACF ENTRY=MVSESA,TYPE=*,REQ=DISP
SYSTEM = KEY7      AUTOMATION CONFIGURATION DISPLAY - ENTRY= MVSESA
-----
AUTOMATION CONFIGURATION DISPLAY - ENTRY= MVSESA
TYPE IS IEE041I
CMD      = (.,'MVS VARY SYSLOG,HARDCPY')
TYPE IS IEE043I
CMD      = (.,'MVS WRITELOG START')
TYPE IS IEE533E
CMD      = (.,'MVS WRITELOG START')
TYPE IS WTOBUF
CODE     = (*,*,,"CANCEL")
END OF MULTI-LINE MESSAGE GROUP

Command ==>
PF1=Help    PF2=End    PF3=Return    PF6=Roll
              PF9=Refresh PF12=Retrieve

```

Figure 43. Display of Automation Control File Settings for Subsystem (DISPACF SUBSYSTEM)

This command displays information for all types of the MVSESA entry, because you accepted the default `TYPE=*`.

DISPAOPS

Purpose

The DISPAOPS command lists the automation operators that are currently active.

Examples

Type `dispaops` on any command line and press the ENTER key. You see a panel similar to the following:

DISPAOPS

A0FK2S0		msys - Command Dialogs		Line 1 of 17	
Domain ID = IPSN7		----- DISPAOPS -----		Date = 06/13/01	
Operator ID = NETOP1				Time = 17:45:05	
System	Automated Function	Primary	Status	Secondary	Status
-----	-----	-----	-----	-----	-----
KEY7	A0FWRK01	AUTWRK01	ACTIV		
KEY7	A0FWRK02	AUTWRK02	ACTIV		
KEY7	A0FWRK03	AUTWRK03	ACTIV		
KEY7	BASEOPER	AUTBASE	ACTIV		
KEY7	GSSOPER	AUTGSS	ACTIV		
KEY7	JESOPER	AUTJES	ACTIV		
KEY7	LOGOPER	AUTLOG	ACTIV		
KEY7	MONOPER	AUTMON	ACTIV		
KEY7	MSGOPER	AUTMSG	ACTIV		
KEY7	MVSCONS	AUTCON	ACTIV		
KEY7	NETOPER	AUTNET1	ACTIV		
KEY7	RECOPER	AUTREC	ACTIV		
KEY7	RPCOPER	AUTRPC	ACTIV		
Command ==>					
PF1=Help		PF2=End	PF3=Return	PF6=Roll	
		PF8=Forward	PF9=Refresh	PF12=Retrieve	

Figure 44. Automation Operators Panel

- The System field shows the name of the system where the automated function is defined
- The Automated Function field shows the name of the automated function used in msys for Operations automation procedures
- The Primary field shows the NVSS automation operator ID assigned to this automated function
- The Status field shows the current status of the primary automation operator
- The Secondary field shows the Backup NVSS automation operator ID assigned to this automated function
- The Status field shows the current status of the backup automation operator

The primary and backup NVSS automation operator IDs are assigned to the automated function in the command dialogs.

DISPASF

Purpose

DISPASF displays the information contained in the automation status file. This includes the:

- Automation status
- Operator ID that last changed the record
- Last threshold exceeded
- Date and time of last monitoring cycle

Examples

If you type `dispasf syslog`, a panel similar to the following is displayed:

```

A0FK3D0X          msys - Command Response      Line 1    of 11
Domain ID   = IPSN7 ----- DISPASF -----    Date = 06/13/01
Operator ID = NETOP1                                     Time = 16:47:09

Command = ASF ID=SYSLOG,REQ=DISP
SYSTEM = KEY7      STATISTICS DISPLAY REQUESTED FOR SYSLOG
-----
STATISTICS DISPLAY REQUESTED FOR SYSLOG
ID= SYSLOG          , TYPE= MVSESA          , STATUS= DOWN
LAST UPDATE BY OPERATOR AUTREC
LAST THRESHOLD EXCEEDED - INFR
OPERATOR NOTIFIED: N
LAST STATUS CHANGE DATE= 06/08/01 , TIME= 10:25 , OPID= AUTREC
LAST MONITORED DATE= 06/08/01 , TIME= 10:25
  ERROR COUNT    DATE      TIME
        01      06/08/01    10:25
        02      06/13/01    12:22
END OF MULTI-LINE MESSAGE GROUP

Command ==>
  PF1=Help      PF2=End      PF3=Return      PF6=Roll
                  PF9=Refresh      PF12=Retrieve

```

Figure 45. Display of Automation Status File Information for DISPASF SYSLOG

DISPATHR

Purpose

The DISPATHR command displays a summary of the msys for Operations thresholds.

Examples

When you type dispathr you see a panel similar to the following:

```

A0FKAAD4          msys - Command Dialogs      Line 1    of 4
Domain ID   = IPSN7 ----- DISPATHR -----    Date = 06/13/01
Operator ID = NETOP1                                     Time = 17:52:14

  System    Resource      Critical    Frequent    Infrequent
  -----
KEY7        DEFAULTS
KEY7        MVSESA
KEY7        SUBSYSTEM
KEY7        SYSLOG        3 in 02:00  2 in 02:00  1 in 02:00

Command ==>
  PF1=Help      PF2=End      PF3=Return      PF6=Roll
                  PF9=Refresh      PF12=Retrieve

```

Figure 46. DISPATHR Command Dialog Panel

- The System field shows the name of the system where the resource is defined.

DISPATHR

- The Resource field shows the name of the resource. The first three entries are always DEFAULTS, MVSESA, and SUBSYSTEM.
- The Critical field shows the critical threshold for the resource.
If no critical threshold has been specified, the field is blank.
- The Frequent field shows the frequent threshold for the resource.
- The Infrequent field shows the infrequent threshold for the resource.
Thresholds have the format *nn* in *hh:mm*, which means that the threshold is exceeded if more than *nn* errors occur within *hh* hours and *mm* minutes.

DISPERRS

Purpose

The DISPERRS command displays information about resources for which errors have been recorded in the status file.

Examples

When you type disperrs you see a panel similar to the following:

AOFKAAD5		msys - Command Dialogs		Line 1 of 9	
Domain ID = IPSN7		----- DISPERRS -----		Date = 06/13/01	
Operator ID = NETOP1				Time = 17:55:23	
System	Resource	Type	Thrs	No	Date Time

KEY7	AOFIPLDT	CONTROL		7	06/06/01 08:09
KEY7				6	04/25/01 15:28
KEY7				5	04/21/01 09:51
KEY7				4	04/20/01 12:42
KEY7				3	04/18/01 17:05
KEY7				2	02/27/01 16:11
KEY7				1	02/16/01 15:15
KEY7	SYSLOG	MVSESA	INFR	2	06/13/01 12:22
KEY7				1	06/08/01 10:25

Command ==>

PF1=HelpPF2=EndPF3=ReturnPF9=RefreshPF6=RollPF12=Retrieve

Figure 47. DISPERRS Command Dialog Panel

- The System field shows the name of the system where the resource is defined.
- The Resource field shows the name of the resource.
- The Type field shows the type of resource.
- The Thrs field shows the type of threshold that has been violated. This is either CRIT (critical), FREQ (frequent), or INFR (infrequent).
- The No field shows the sequence number assigned to the error.
- The Date and Time fields show the date and time when the error occurred.

DISPFLGS

Purpose

The DISPFLGS command shows whether msys for Operations functions are enabled (Recovery flag set to Y), or disabled (Recovery flag set to N).

Examples

When you type dispflgs in msys for Operations you see a panel similar to the following:

AOFKAAU	msys - Command Dialogs										Line 1 of 8		
Domain ID = IPSN7	----- DISPFLGS -----										Date = 06/13/01		
Operator ID = NETOP1											Time = 18:02:59		
System = KEY7													
Resource	Actual					Effective					Settings		
	A	I	S	R	D RS	A	I	S	R	D RS			
-----											-----		
DEFAULTS	Y	-	-	-	-	Y	Y	Y	Y	Y	- No explicit setting		
MVSESA	-	-	-	-	-	Y	Y	Y	Y	Y	N Turned off		
CDS	-	-	-	Y	-	Y	Y	Y	Y	Y	E Consult exit		
LOG	-	-	-	Y	-	Y	Y	Y	Y	Y	Y Turned on		
LOGGER	-	-	-	N	-	Y	Y	Y	N	Y	? Error		
WTO	-	-	-	Y	-	Y	Y	Y	Y	Y			
XCF	-	-	-	Y	-	Y	Y	Y	Y	Y	Flags		
SUBSYSTEM	-	-	-	-	-	Y	Y	Y	Y	Y	-----		
												A Automation	
												I Initial start	
												S Start up	
												R Recovery	
												D Shut down	
												RS Restart	
Command ==>													
PF1=Help			PF2=End			PF3=Return			PF6=Roll				
						PF9=Refresh			PF12=Retrieve				

Figure 48. DISPFLGS Command Dialog Panel of msys for Operations

The resource column shows whether the WTO buffer shortage and LOG recovery are switched on.

DISPMSGGS

Purpose

The DISPMSGGS command displays which automation operators receive each automated message.

Examples

Type dispmsggs on a command line to display the Authorized Message Receivers panel:

DISPMSGSGS

AOFK2SM			msys	-	Command Dialogs	Line 1	of 9
Domain ID	=	IPSN7	----- DISPMSGSGS -----			Date =	06/13/01
Operator ID	=	NETOP1	System = KEY7			Time =	18:08:09
Message	Primary Receivers			Secondary Receivers			

'AOF*'	AUTMSG	AUTSYS	AUTBASE				
'IEA*'	AUTREC	AUTSYS	AUTBASE				
'IEE889I	AUTSYS	AUTBASE					
'IEE*'	AUTREC	AUTSYS	AUTBASE				
'IOS*'	AUTREC	AUTSYS	AUTBASE				
'IXC263I	AUTXCF2	AUTSYS	AUTBASE				
'IXG257I	AUTXCF2	AUTSYS	AUTBASE				
'IXG261E	AUTXCF2	AUTSYS	AUTBASE				
'*'	AUTLOG	AUTSYS	AUTBASE				
Command ==>							
PF1=Help		PF2=End		PF3=Return		PF6=Roll	
				PF9=Refresh		PF12=Retrieve	

Figure 49. Authorized Message Receivers Panel

- The Message field shows the message or message prefix.
- The Primary Receivers field shows the automation operators, identified by their NVSS IDs. The Primary Receivers column lists automation operators that can receive the messages listed beside their names. These messages go to the first automation operator listed in the Primary Receivers column that is active.
- The Secondary Receivers field shows the Alternate automation operators, identified by their NVSS IDs. Secondary Receivers receive copies of the messages listed beside their names.

DISPWTOR

Purpose

The DISPWTOR command displays all WTORs that are currently outstanding.

Examples

If you type dispwtor you see a panel similar to the following:

AOFKADAC		msys - Command Dialogs		Line 1 of 2
Domain ID = IPSN7		----- DISPWTOR -----		Date = 06/13/01
Operator ID = NETOP1				Time = 18:15:12

Rply	System	Subsystem	Message
-----	-----	-----	-----
32	KEY7	MVSESA	DSI802A IPSN7 REPLY WITH VALID NCCF SYSTEM OPERATOR COMMAND

Command ==>			
PF1=Help	PF2=End	PF3=Return	PF6=Roll
		PF9=Refresh	PF12=Retrieve

Figure 50. Display of Outstanding Replies (DISPWTOR)

- The Rply field shows the outstanding reply number.
- The System field shows the name of the system for which the reply is outstanding.
- The Subsystem field shows the subsystem name, or MVSESA if the reply is from a resource that is not a subsystem.
- The Message field shows the text of the message.

INGAUTO

Purpose

INGAUTO ON MVSESA.*function_qualifier* enables an msys for Operations function.

INGAUTO OFF MVSESA.*function_qualifier* disables an msys for Operations function.

Examples

To turn all automation on for the WTOR buffer shortage recovery, type the following:

```
ingauto on,mvsesa.wto
```

INGCUST

Purpose

INGCUST is a service function that is used in msys for Operations for checking the syntactical correctness of the customization member AOFECUST.

After changing AOFECUST in msys for Operations, check the syntactical correctness of the member before starting msys for Operations. If AOFECUST is syntactically incorrect, msys for Operations cannot be started. To perform the check issue INGCUST with INSTORE as its first parameter. INGCUST informs you about the result with a message.

INGCUST

INGCUST does not require that the name of the input member is AOFCUST. But in order to use a member with a different name for the customization of msys for Operations, you must rename it back to AOFCUST.

Examples

```
INGCUST INSTORE MYCUST
```

This command checks the customization member MYCUST for syntactical correctness. Note that in order to use it, you must rename it to AOFCUST.

Part 5. Setup reference

Chapter 16. msys for Operations Definition Statements Reference

Definition statements are used by msys for Operations for performing system administration tasks. System administration is the process of redefining system defaults and storage requirements. You can perform administration subtasks during msys for Operations installation or when you redefine network resources.

In planning for installation and network management tasks, you determine the facilities you need to run msys for Operations. You also need to identify hardware requirements and specific resources used by msys for Operations. This information can help you determine requirements for user coding and msys for Operations definitions.

After copying current definition statements from the sample files, you can begin to modify existing definition statements or create new ones. You can alter definition statements during a first-time msys for Operations installation, or later, while running your production system.

Statement Formats

The format of a definition statement is:

- Statement name
- General introduction

The general introduction explains overall options, assumptions, and the purpose of the statement. Each introduction explains the name of the member and where you code the statement.

- Definition statement syntax

The definition statement syntax is a model statement that is formatted according to the code conventions.

- Operand descriptions

This section describes each operand you can specify for the definition statement. The description includes the specific values or variable information that you can specify for the operand.

Syntax Conventions for Definition Statements

These syntax conventions apply to most statements:

- Code at least one blank between a label name and the name of the definition statement, and between the name of the definition statement and the first operand. One or more blanks, or a single comma with no blanks, must separate the statement operands. You cannot separate the operands with a combination of commas and blanks. If you omit the optional label name, you still need to precede the definition statement with one or more blanks.
- The label field must not exceed 8 characters, and the field must start in column 1.
- Continuation from one line to the next is not allowed. However, you can repeat the definition statement and add the remaining information. For the following example:

```
LOGINIT AUTOFLIP=YES
LOGINIT RESUME=YES
```

Is the same as:

```
LOGINIT AUTOFLIP=YES,RESUME=YES
```

- Place comments on a separate line for DSIPARM members. The first column of a comment line must contain an asterisk (*).
- Many definition files conclude with an END statement. This END statement has no operands and cannot begin in column 1.
- All NVSS program identifiers, which are called names, must not exceed 8 characters unless specified. The first character must be alphabetical and alphabetical characters must be in uppercase.
- Command names, command list names, and any other NVSS program identifiers must not contain commas (,), periods (.), blanks (), apostrophes ('), ampersands (&), asterisks (*), or equal signs (=). Commas, periods, blanks, and equal signs are used as delimiters when the definition statements are parsed. The other characters have special meanings for NVSS command lists.
- Command names and command list names must begin in column 1.
- System symbolics can be coded on any NVSS definition statement to provide unique information to the msys for Operations system. System symbolics are useful when running msys for Operations on different systems where you want to have different characteristics. This unique information (as defined by the system symbolic values) will remain on your system definitions until you change those definitions and re-IPL MVS.

AUTH

The AUTH statement defines an operator's authority to view and control resources and specifies whether an operator is eligible to be the authorized receiver. Code this statement in a member specified by a PROFILEN statement associated with the operator. See "OPERATOR" on page 161 and "PROFILEN" on page 170 for information on how a PROFILEN statement is associated with an operator. A sample member supplied with msys for Operations is DSIPROFA.

The syntax for the AUTH statement is:

AUTH



Where:

MSGRECVR=NO|YES

Specifies whether operators using this profile can receive unsolicited messages that are not routed to a particular operator by the use of the NVSS ASSIGN command or by NVSS automation.

NO

Indicates that operators using the profile containing this statement do not receive unsolicited messages. NO is the default.

YES

Indicates that an operator using this profile can be the authorized message receiver.

Usage Notes:

- In NVSS, the *authorized receiver* is the operator authorized to receive all the unsolicited and authorized messages that are not routed to a specific operator with an ASSIGN command or a ROUTE action in an NVSS automation statement. The authorized receiver is determined by the order in which operator terminals are defined to NVSS and by the order in which authorized operators have logged on.
- When several operators are eligible to receive a particular message, NVSS uses the following priority order (from the lowest to the highest) to route the message to the proper operator:
 - The operator designated by an ASSIGN command
 - The operator or operators designated by the ROUTE action in the automation table
 - An operator

If more than one operator is logged on, the one logged on first has priority.
 - An autotask operator

If more than one autotask has been started, the one started first has priority. Use the ASSIGN command if an autotask is going to be the receiver of unsolicited messages.
 - The system console operator

OPERATOR

When OPERSEC is not specified as SAFDEF on either the OPTIONS statement in DSIDMN or on the REFRESH command, the OPERATOR statement identifies each operator who can log on to this NVSS program. This statement is also used to define operator identifiers that can be started as automation tasks by the AUTOTASK command. The OPERATOR statement must come before its associated PROFILEN statements. You code this statement in DSIOPF.

You can dynamically add or delete operators by adding or deleting OPERATOR statements in DSIOPF and issuing the REFRESH OPERS command.

The syntax for the OPERATOR statement is:

OPERATOR

```
►► opid OPERATOR PASSWORD=password [ , NOCHECK ] ◀◀
```

Where:

opid

Indicates the 1–8 character value that identifies an operator. Valid characters for the operator identifier are letters A–Z, the numbers 0–9, or the special characters number sign (#), at sign (@), or dollar sign (\$). The identifier must begin in column 1. Each operator must have a unique operator identifier. Also, code an operator statement for each operator identifier you want to use for an automation task. Do not use the names of hard-copy logs, terminals, or task

OPERATOR

identifiers as operator identifiers. The following identifiers are reserved by the NVSS program and cannot be used as operator identifiers:

- ALL
- DPR
- DST
- HCL
- HCT
- LOG
- MNT
- NNT
- OPT
- OST
- PPT
- SYSOP
- TCT

Additionally, if the operator identifier is the same as the LU name (terminal), some command lists assume that the operator is an autotask and do not run.

PASSWORD=*password*

Indicates the 1–8 character operator password. You are required to code a password, but the password is ignored if you code OPTIONS OPERSEC=MINIMAL or OPERSEC=SAFCHECK in DSIDMN. The password is also ignored if you use this operator identifier when starting an autotask using the AUTOTASK command. For operator identifiers set up specifically for autotasks, use the password to identify the operator as such.

NOCHECK

Allows the NVSS operator to log on without NVSS verifying the password. In this case, message DWO354 is sent to the authorized receiver indicating that the operator has logged on and the password has not been verified by NVSS. The advantage of this option over OPERSEC=MINIMAL is that the operator's profile is used and any initial command specified is used. NOCHECK must be preceded by a comma.

OPTIONS

The OPTIONS statement describes the type of security settings msys for Operations uses. This checking includes:

- Authorizing a user to log on as an msys for Operations operator
- Validating the password and identity of a user logging on to msys for Operations
- Specifying where the attributes for an msys for Operations operator are defined
- Protecting commands executed in NVSS

Code the OPTIONS statement in DSIDMN. You can use the REFRESH command to dynamically change these options.

There can be more than one OPTIONS statement. If you specify the same keyword on more than one OPTIONS statement, the first occurrence is used. Subsequent specifications of a keyword result in an error message and initialization continues.

During initialization, messages BNH180I, BNH191I, and BNH193I are issued to display the operator and command security settings that will be used by msys for Operations. If error messages are displayed during initialization, message DSI813A is issued at the end of initialization to give the operator a chance to continue or to terminate initialization.

OPTIONS

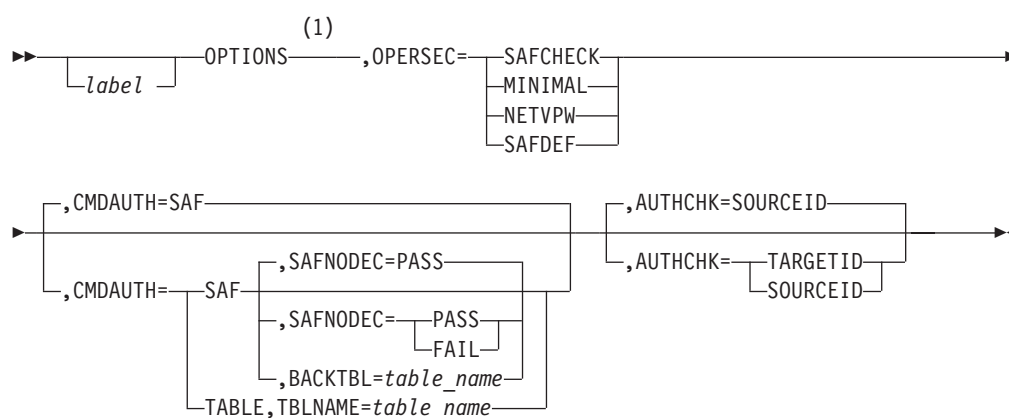
An **OPTIONS** statement is required for **msys** for Operations and a value must be specified for the **OPERSEC** and **CMDAUTH** keywords.

If OPERSEC=SAFDEF, you do not need a DSIPRF specification in your msys for Operations procedure. If you later want to issue a REFRESH command with OPERSEC specified as either NETVPW or SAFCHECK, you must first dynamically allocate DSIPRF, if it is not in your msys for Operations procedure.

If OPERSEC=SAFDEF, you do not need a DSIOPF member in DSIPARM. If you later want to issue a REFRESH command with OPERSEC specified as either NETVPW or SAFCHECK, DSIPARM must contain a DSIOPF member for the REFRESH command to complete successfully.

The syntax for the `OPTIONS` statement is:

OPTIONS



Notes:

- 1 For more information about the interrelationship of keywords, see Table 9 on
page 167.

Where:

label

Indicates the optional label for the `OPTIONS` statement. This label identifies the statement for any related error messages.

OPERSEC=MINIMAL | NETVPW | SAFCHECK | SAFDEF

Defines the method used to allow users to log on to NVSS.

MINIMAL

Specifies that msys for Operations operators are defined by a list of operator identifiers in DSIOPE. There is no password validation. The logon profile is not used and logon operands specified on the logon menu are ignored.

Other keywords cannot be specified on the OPTIONS statement when MINIMAL is specified. If other keywords are specified, they are ignored, a message is issued, and initialization continues. Message DSI813A is issued at the end of initialization, indicating these errors and giving the operator a chance to continue or to terminate initialization.

OPTIONS

You cannot use the REFRESH command to change the OPERSEC specification when MINIMAL is specified.

NETVPW

Specifies that msys for Operations operators are defined by a list of operator identifiers in DSIOPF. The identification is validated with a password associated with the identifier in DSIOPF. The profile, read from DSIPRF at logon, contains information about what the operator is allowed to do, and limits commands and resources that the operator can use.

CMDAUTH=SAF should not be specified when OPERSEC=NETVPW. If CMDAUTH=SAF is specified, no authorization checks are performed when commands are issued.

If an error with the CMDAUTH specification is detected, an error message is issued and initialization continues. Message DSI813A is issued at the end of initialization, indicating that errors have occurred and giving the operator a chance to continue or to terminate initialization.

SAFCHECK

Specifies that operator identification and password checking is performed using an SAF security product. The operator identifier must also be defined in DSIOPF, and other attributes given to the operator at logon are taken from the operator's specified profile in DSIPRF.

Security access checks that occur when an operator tries to access a data set that is protected in the DATASET class of an SAF product or an MVS system command that is protected in the OPERCMDS class of an SAF product are checked against the authority of the operator.

This specification allows CMDAUTH to be specified as TABLE or SAF.

SAFDEF

Specifies that operator identification and password checking is done using an SAF security product. Authority to log on as an msys for Operations operator is controlled through the APPL class. The operator identifier must be authorized to the resource name in the APPL class which represents msys for Operations.

The attributes given to the operator at logon are defined in the NVSS segment of the operator's user profile in the SAF product.

When OPERSEC=SAFDEF is specified, any value for CMDAUTH can be used.

CMDAUTH=SAF | TABLE

Defines the method used by msys for Operations to protect command usage. For a list of commands with keywords and values that can be protected, see "Recommended commands to protect" on page 55.

CMDAUTH cannot be specified when OPERSEC=MINIMAL. With other OPERSEC settings you can issue the REFRESH command to change the method used for command authorization.

If the CMDMDL statement for a command specifies SEC=BY, no authority checking is done for that command. Command authorization checking for automation table commands can also be bypassed by specifying AUTOSEC=BYPASS on the DEFAULTS command.

SAF Specifies that msys for Operations performs command authorization

checking using an SAF security product. The commands you want to protect should be defined in the NETCMDS class in the security product. The operators that you want to give access to the commands can be permitted to use the resource names that represent commands, keywords, and values. For more information, see “Command authorization” on page 51.

If AUTHCHK is not specified when CMDAUTH=SAF, it defaults to SOURCEID during initialization. When the REFRESH command is used, there is no default for AUTHCHK.

If, during initialization, the NETCMDS class is not active or the security product is not active and BACKTBL is not specified, no authorization checking is performed when commands are issued. An error message is issued and initialization continues. Message DSI813A is issued at the end of initialization, indicating that errors have occurred and giving the operator a chance to continue initialization or to terminate it.

If, during command authorization checking, the NETCMDS class is inactivated, the security product is inactivated, or there is no resource name defined for the command being issued, msys for Operations uses the command authorization table specified by BACKTBL. If BACKTBL is not specified, authority to issue the command is determined by the value of SAFNODEC.

For immediate commands, authority checking is not performed by SAF. These commands should be protected using the command authorization table specified by BACKTBL.

CMDAUTH=SAF can only be specified when OPERSEC=SAFCHECK or OPERSEC=SAFDEF. If CMDAUTH=SAF is specified with any other value for OPERSEC, initialization continues, and no authorization is in effect. Message DSI813A is issued at the end of initialization, indicating that errors have occurred and giving the operator a chance to continue or to terminate initialization.

TABLE

Specifies that authorization checking is performed using a dynamic command authorization table specified by the TBLNAME keyword. The table can be modified and reloaded using the REFRESH command.

If AUTHCHK is not specified, it defaults to SOURCEID for CMDAUTH=TABLE during initialization. When the REFRESH command is used, there is no default for AUTHCHK.

If CMDAUTH=TABLE is specified and TBLNAME is not specified, or if the member specified by TBLNAME is not found, messages are issued and command authorization checks are not performed. Message DSI813A is issued at the end of initialization, indicating that errors have occurred and giving the operator a chance to continue initialization or to terminate it.

BACKTBL=*table_name*

Specifies the 1–8 character name of the command authorization table used when CMDAUTH=SAF and SAF checking cannot be performed for the command being issued. This can occur when:

- The command is an immediate command.
- There is no resource name defined in the NETCMDS class which protects or authorizes this command.

OPTIONS

- The NETCMDS class is not active.
- The security product is not active.

The table name is a member name in a DSIPARM data set.

The BACKTBL and SAFNODEC options are mutually exclusive. When both are specified, the BACKTBL option is used.

table_name

Specifies the name of the table. For more information on how to build the table, see “Command authorization” on page 51.

If CMDAUTH=SAF is specified and the backup command authorization table contains syntax errors, messages are issued. Message DSI813A is issued at the end of initialization, indicating that errors have occurred and giving the operator a chance to continue initialization or to terminate it. If the reply to DSI813A is YES, the table will still be loaded with the statements that are coded correctly. If the backup table contains no valid statements, NVSS continues with no backup table.

AUTHCHK=SOURCEID | TARGETID

Specifies the user ID that is to be used when verifying command authorization and span authorization for VTAM commands. For specific information about how the SOURCEID and TARGETID are determined, “Command authorization” on page 51.

SOURCEID

Specifies to check the authority of the original issuer or the ID closest to the original issuer.

Access failure messages display the source issuer of the command. SOURCEID is the default when CMDAUTH=SAF.

TARGETID

Specifies to check the authority of the task under which the command runs.

SAFNODEC=PASS | FAIL

Defines the action taken by msys for Operations when BACKTBL is not specified and the SAF product cannot make a decision on command authority. This can occur when:

- There is no resource name defined in the NETCMDS class which protects or authorizes this command.
- The NETCMDS class is not active.
- The security product is not active.

The default is to pass the command.

The SAFNODEC and BACKTBL options are mutually exclusive. When both are specified, the BACKTBL is used.

PASS Specifies that users should be allowed to issue all commands when the SAF product cannot make a security decision.

FAIL Specifies that users should not be allowed to issue any commands when the SAF product cannot make a security decision.

Attention: Specifying PASS indicates that no command authorization checking is performed if the SAF product becomes unavailable or otherwise cannot make a decision. Specifying FAIL keeps you from issuing

any commands (except when SEC=BY is specified on the CMDMDL statement or AUTOSEC=BYPASS is in effect for automation commands) if the SAF product becomes unavailable or otherwise cannot make a decision. For this reason, it is better to define a backup command authorization table than to use SAFNODEC.

Interrelationships of Keywords:

Table 9 shows the relationships between OPERSEC and CMDAUTH specifications.

Table 9. Interrelationships between OPTIONS Keyword Values (Part 1 of 2)

IF OPERSEC=	CMDAUTH	
	TABLE	SAF
MINIMAL	Not valid	Not valid
NETVPW	Is valid	Not valid
SAFCHECK	Is valid	Is valid
SAFDEF	Is valid	Is valid

Table 10 shows the relationships between CMDAUTH, TBLNAME, BACKTBL, SAFNODEC, and AUTHCHK specifications.

Table 10. Interrelationships between OPTIONS Keyword Values (Part 2 of 2)

CMDAUTH=	TBLNAME	BACKTBL	SAFNODEC	AUTHCHK	
				TARGETID	SOURCEID
SAF	Ignored	Is valid	Is valid, defaults to PASS, and is ignored if BACKTBL is specified	Is valid	Default
TABLE	Required	Ignored	Ignored	Is valid	Default

Operator Authority:

Table 11 shows how the OPERSEC and OPSPAN keywords are used to specify how operator verification and authority checking is to be performed.

Table 11. Defining and Verifying Operator Authority

Keyword	Related Defaults	Restrictions	Effect
OPERSEC= MINIMAL		<ul style="list-style-type: none"> CMDAUTH is ignored. Cannot use REFRESH to change OPERSEC. 	<ul style="list-style-type: none"> Logon profile ignored Logon operands ignored No password validation Operator must be defined in DSIOF
OPERSEC= NETVPW	CMDAUTH=TABLE must be specified.	<ul style="list-style-type: none"> Not valid when CMDAUTH=SAF 	<ul style="list-style-type: none"> Password validated from DSIOF Operator must be defined in DSIOF Profile read from DSIPRF

OPTIONS

Table 11. Defining and Verifying Operator Authority (continued)

Keyword	Related Defaults	Restrictions	Effect
OPERSEC= SAFCHECK	<ul style="list-style-type: none"> CMDAUTH must be specified. 		<ul style="list-style-type: none"> Password verification using SAF product Operator must be defined in DSIOPF Profile read from DSIPRF NVSS task user IDs are used for any SAF calls for NVSS operators, such as to the DATASET class.
OPERSEC= SAFDEF	<ul style="list-style-type: none"> CMDAUTH must be specified. 		<ul style="list-style-type: none"> Password verification using SAF product Operator logon authority using RACF APPL class Operator attributes defined in NetView segment of SAF product NVSS task userids are used for any SAF calls for NVSS operators, such as to the DATASET class.

Protecting Commands:

Table 12 shows how the CMDAUTH and related options are used to specify how operator command authority checking is to be performed.

Table 12. Protecting Commands Executed in NVSS

Keyword	Related Defaults	Restrictions	Effect
CMDAUTH= TABLE	AUTHCHK default is SOURCEID.		Command authorization is based on the specified table.
CMDAUTH= SAF	AUTHCHK default is SOURCEID.	Cannot be specified if OPERSEC is MINIMAL or NETVPW.	Command authorization using the NETCMDS class of an SAF product. Immediate commands are not checked in the NETCMDS class, but a backup command authorization table can be used for this purpose.
TBLNAME= <i>table_name</i>		Required if CMDAUTH=TABLE	Identifies the table to be used

Table 12. Protecting Commands Executed in NVSS (continued)

Keyword	Related Defaults	Restrictions	Effect
BACKTBL= <i>table_name</i>		Valid only when CMDAUTH=SAF	Specifies the backup table to be used for immediate commands and when the SAF product cannot make a security decision. This can occur when: <ul style="list-style-type: none"> • There is no resource name defined in the NETCMDS class which protects or authorizes this command • The NETCMDS class is not active • The security product is not active.
SAFNODEC= PASS FAIL		Defaults to PASS Only used when CMDAUTH=SAF and no BACKTBL is specified	Identifies whether to PASS or FAIL command authority checking if the SAF product can reach no decision.
AUTHCHK= SOURCEID	Default when CMDAUTH=TABLE	Authorization is based on the authority of the original issuer of the command or the ID closest to the original issuer.	
AUTHCHK= TARGETID			Authorization is based on the authority of the ID under which the command runs.

Usage Notes:

- When CMDAUTH=SAF and no backup command authorization table is used, immediate commands are not checked and will pass even if SAFNODEC=FAIL.

PROFILE

The PROFILE statement defines the profile name to the system. PROFILE must be the first statement in each profile definition. Code this statement in a member specified by a PROFILEN statement associated with the operator. See “OPERATOR” on page 161 and “PROFILEN” on page 170 to determine how a PROFILEN statement is associated with an operator. Profiles are not used when OPERSEC=SAFDEF. Examples of sample members supplied with the NVSS program are DSIPROFA and DSIPROFB.

The syntax for the PROFILE statement is:

PROFILE

```

>>—profilename PROFILE—┐
                           └─HCL=hclname─┐
                                           └─,CONSNAME=consname─┐
                                                                     └─,IC=text─┐

```

Where:

PROFILE

profilename

Indicates the name of the member that contains the profile. This name must begin in column 1.

HCL=*hclname*

Indicates the name of the hardcopy printer that is automatically started when this operator logs on. Define this name in the VTAM definition and in the msys for Operations HARDCOPY definition statement in DSIDMN. HCL is an optional operand. The IC keyword, when specified, must always be specified as the last keyword.

Although each operator can be assigned to only one hardcopy printer, several operators can share the same printer. However, if too many operators share the same hardcopy printer, messages for that device can accumulate and messages might not be printed for some time after they are received.

CONSNAME=*consname*

Indicates the default extended console name for operators using this profile. This default console name is used when the operator does not specify a console name using the GETCONID or SETCONID command. It is also the console name used when you issue the MVS command and have not previously obtained an extended console. The console name must be a 2- to 8-character value, as required by MVS. Valid characters for console names are A-Z, 0-9, @, #, or \$. The first character of the console name must be alphabetic (A-Z) or one of the following special characters: @, #, or \$. For more information on console names, refer to the MVS library. For more information on the implications of specifying CONSNAME, refer to the GETCONID and SETCONID command in the msys for Operations online help. The IC keyword, when specified, must always be specified as the last keyword.

IC=*LOGPROF1|LOGPROF2|LOGPROF3|LOGPROF4*

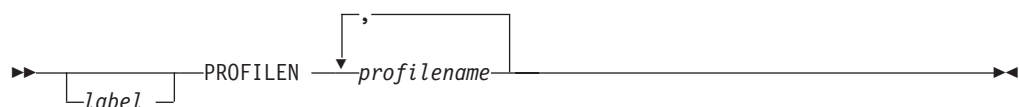
Specifies the command list that is run immediately after a successful log on. Use LOGPROF1 for Msys for Operations operators, because it defines PF keys and a unique console name. Use LOGPROF2 for the AUTO1 autotask. LOGPROF3 assigns a unique console name to the task and should be used for the Msys for Operations autotasks other than AUTO1 and AUTO2. Use LOGPROF4 for the AUTO2 autotask.

PROFILEN

The PROFILEN statement associates the name of a particular profile or list of profiles with an operator identification. Code PROFILEN as often as necessary to ensure that all the possible profile names are associated with a particular operator identification. An OPERATOR statement must precede each PROFILEN statement or group of statements. You code this statement in DSIOPF. Profiles are not used when OPERSEC=SAFDEF.

The syntax for the PROFILEN statement is:

PROFILEN



Where:

label

Indicates the optional label for the PROFILEN statement. This label identifies the statement in any related error messages.

profilename [...]

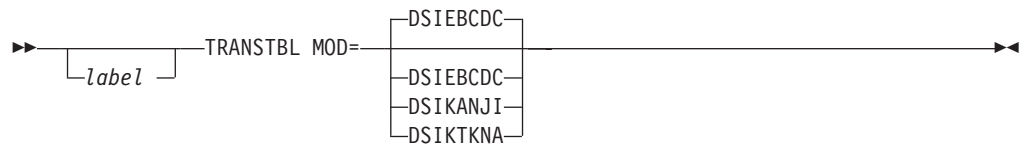
Indicates the profile name to be associated with the operator identification defined in the preceding OPERATOR statement. The profile name is a 1–8 character name that matches the *profilename* given on a PROFILE statement in a profile member. The first name listed in the first PROFILEN statement is used by default if an operator does not specify a *profilename* in the log on request. Note that multiple profile names must be separated by commas.

TRANSTBL

The TRANSTBL statement defines a character translation set to the NVSS program. Code this statement in the DSIDMN member. Stop and restart the NVSS program to implement the changes.

The syntax for the TRANSTBL statement is:

TRANSTBL



Where:

label

Is the optional label for the TRANSTBL statement. This label identifies the statement in any related error messages.

MOD=DSIEBCDC | DSIKANJI | DSIKTKNA

Specifies a particular load module name that contains a 1024-byte character translation set. Specify MOD=DSIEBCDC for EBCDIC support, MOD=DSIKANJI for kanji support, or MOD=DSIKTKNA for katakana support.

DSIEBCDC

Selects an 8-bit coded character set called EBCDIC. This is the default.

DSIKANJI

Selects a character set of symbols used in Japanese ideographic alphabets called kanji.

DSIKTKNA

Selects a Japanese character translation set called katakana.

Usage Notes:

- All devices must use the same character set for meaningful results.
- The TRANSTBL value for the log printer program should be the same value as the value used for the NVSS program definition. The NVSS program does not check these values for compatibility.
- If you define more than one TRANSTBL statement, the NVSS program uses the last one you entered.

TRANSTBL

Part 6. Automation-related messages

Chapter 17. Messages

AOF204I *time* : EXPECTED PARAMETERS
MISSING OR INVALID FOR REQUEST
clist_name - *parameter_name*

Explanation: One or more parameters that must be passed to the command list identified in this message are missing or are not valid.

The variable *time* shows the time this message was generated.

The variable *clist_name* shows the name of the command list to which the variables must be passed.

The variable *parameter_name* shows the name of the parameter that was missing or not valid.

System Action: None.

Operator Response: Make sure that the syntax is correct and enter the request again. The netlog will show both the message and the request as processed. Notify the system programmer if the request fails again.

System Programmer Response: Review the user input to determine the validity of the request and instruct the operator on the proper procedure. If the user input is correct, set debugging on for the command list to determine the source of the error.

Classes: 0.

AOF900I AT LEAST ONE STRUCTURE IS STILL
ALLOCATED IN THE CF *cfname*

Explanation: The process was started disconnecting the coupling facility *cfname* from its connected systems. However, at least one structure that is still allocated was detected by the DRAIN function. Or, one allocated structure that has a connection to an application was detected by the FORCE function.

System Action: Processing terminates.

Operator Response: Check the allocated structures of the coupling facility. Rebuild the structures to another coupling facility if possible before issuing the command again.

System Programmer Response: None.

Classes: 40, 43.

Modules: INGRX912, INGRX913

AOF901I PATH *chpid* FROM SYSTEM TO CF
cf_name COULD NOT BE SET TO *status*

Explanation: The program received an unexpected message in response to the CF CHP command setting a

sender path from a connected system to a coupling facility to either OFFLINE or ONLINE.

System Action: Processing terminates.

Operator Response: None.

System Programmer Response: Check the netlog for related messages to determine the cause of the error.

Classes: 40 43.

Modules: INGRX913, INGRX915

AOF902I DRAINING THE CF *cf_name* DID NOT
COMPLETE SUCCESSFULLY

Explanation: The program received an unexpected message in response to the CF CHP command setting a sender path from a connected system to a coupling facility to OFFLINE.

System Action: Processing terminates.

Operator Response: None.

System Programmer Response: Check the netlog for related messages to determine the cause of the error.

Classes: 40 43.

Modules: INGRX913

AOF903I EITHER THE SPECIFIED CF IS NOT
DEFINED OR NO CF HAS BEEN
DEFINED

Explanation: Either the command was executed in a sysplex that does not use any coupling facility or the name of the coupling facility is not known in the sysplex.

System Action: Processing terminates.

Operator Response: Specify a valid coupling facility name.

System Programmer Response: None.

Modules: AOFRVX2A, AOFRVX22, AOFRX210, INGRX900, INGRX901

AOF904I A CF/CDS PROCESS IS STILL
RUNNING. ACTION *action name* IS
REJECTED

Explanation: While the AUTXCF task is busy, new actions are rejected. This avoids performance degradation due to multiple rebuild processes, or unpredictable results due to multiple executions of an action.

The variable *action* shows the CF or CDS action that is being performed.

The variable *name* shows the name of the CF or the CDS.

System Action: Processing terminates.

Operator Response: Retry the action later.

System Programmer Response: None.

Modules: INGRVX3D, INGRVX90

AOF905I *action* OF STRUCTURE *strname* FAILED

Explanation: The program received an unexpected message in response to a SETXCF command. The action that is related to the structure did not complete successfully.

System Action: Processing terminates except when all structures of a coupling facility are being rebuilt. In this case processing continues with the next structure.

Operator Response: None.

System Programmer Response: Check the netlog for related messages to determine the cause of the error.

Classes: 40 43.

AOF906I TIME-OUT OCCURRED ON
COMMAND *command*

Explanation: The MVS command did not respond in time. Especially XCF commands can timeout more often because they are processed by one single task.

System Action: Processing terminates except when all structures of a coupling facility are being rebuilt. In this case processing continues with the next structure.

Operator Response: Before executing the action again, check the netlog and the system activities to determine the cause of the timeout.

System Programmer Response: None.

Classes: 40 43.

AOF907I STRUCTURE *strname* HAS
CONNECTIONS OTHER THAN
FAILED-PERSISTENT

Explanation: The automation detected at least one connection to the structure that does not have the status FAILED-PERSISTENT. Because XCF does not allow for the deletion of such a structure, the action is rejected.

System Action: Processing terminates.

Operator Response: Make sure that the structure has either no connections, or only FAILED-PERSISTENT connections before you force the deletion of the structure again.

System Programmer Response: None.

Classes: 40 43.

Modules: INGRX911

AOF908I START OF AUTOTASK *taskname*
FAILED. RC = *rc*

Explanation: Any action that is related to a sysplex resource, for example, a coupling facility, is performed on a dedicated autotask. If the task is not active, it is started via the AUTOTASK command. However, the command failed with the return code *rc*.

System Action: Processing terminates.

Operator Response: Inform your system programmer.

System Programmer Response: Analyze the return code and correct the error. Refer to the description of the AUTOTASK command help in the online help (HELP AUTOTASK).

Classes: 40 44.

Module: INGRVX90

AOF909I FUNCTION *function* REJECTED DUE
TO A CF STATUS CHANGE

Explanation: You requested the function *function* to be performed. Meanwhile the status of the affected coupling facility has changed. The function is rejected because the current status of the coupling facility does not match the status required by the function.

System Action: The program terminates.

Operator Response: Press the refresh key to display the latest status.

System Programmer Response: None.

Classes: None.

Module: INGRVX90

AOF910I POLICY *policy* COULD NOT BE
ACTIVATED

Explanation: During verification of the policy *policy* the automation detected that the policy does no longer fulfill the criteria to become active. The automation rejected the command to make the policy the active policy.

System Action: Processing terminates.

Operator Response: Refresh the screen. Then retry the operation.

System Programmer Response: None.

Classes: None.

Module: INGRVX90

AOF911I POLICY CHANGE(S) PENDING

Explanation: After the new policy has become active, the automation detected that one or more structures have a policy change pending.

System Action: The command completed successfully.

Operator Response: Use the command INGCF STR to find out what structures have a pending status. Rebuild each of these structures to remove its pending status.

System Programmer Response: None.

Classes: None.

Module: INGRVX90

**AOF912E IMPROPER SHARE OPTIONS
(*shroption*) DETECTED FOR LOG
STREAM *logstream***

Explanation: At least the current VSAM data set of the log stream *logstream* has an incorrect share option (other than '1,3'). This may cause unnecessary allocations of log stream data sets. When retrieving or deleting log records or connecting to the log stream, return codes indicating gap(s) will be issued.

System Action: Processing continues.

Operator Response: Inform your system programmer.

System Programmer Response: Check the VSAM definitions of the affected log stream including the data class being used. Correct the share option accordingly.

Classes: 40, 43.

Module: INGRX710

**AOF913I *itemname* VALUE OF CDS TYPE *type*
HAS BEEN CHANGED FROM *oldvalue*
TO *newvalue***

Explanation: The automation detected a situation where the size of a couple data set must be increased to prevent an outage situation. The value of the item *itemname* has been increased as stated.

System Action: Processing continues.

Operator Response: None.

System Programmer Response: Determine the reason for the change and take action accordingly.

Classes: 40, 43.

Module: INGRX711

**AOF914E RECOVERY OF THE CDS TYPE *type*
FAILED. RSN= *reason***

Explanation: The recovery of the couple data set of type *type* failed.

Reason Explanation

- 10 The command 'SETXCF COUPLE,PSWITCH' failed.
- 19 The LOGR couple data set(s) could not be extended.
- 20 The control card information of the primary couple data set could not be obtained.
- 21 A temporary couple data set with the new item values could not be formatted.
- 22 The temporary couple data set could not be allocated.
- 23 The old alternate couple data set could not be deleted.
- 24 The temporary couple data set could not be made the primary couple data set.
- 25 The old primary couple data set could not be deleted.
- 26 The old primary couple data set could not be reformatted.
- 27 The reformatted old primary couple data set could not be allocated as the new alternate couple data set.
- 28 The reformatted old primary couple data set could not be made the new primary couple data set.
- 29 The temporary couple data set could not be deleted.
- 2A The old alternate couple data set could not be reformatted.
- 2B Either the reformatted old alternate couple data set or a new spare couple data set could not be allocated as the alternate couple data set.

System Action: Processing terminates.

Operator Response: Inform your system programmer.

System Programmer Response: Check the netlog and the system log for related system messages. Determine the reason and take action accordingly.

Reason Explanation

- 10 Check the system log to determine the reason why XCF did not switch the couple data sets. If the message was issued while extending the couple data sets, follow the comment for reason codes 20 through 29.
- 19 Follow the description of reason codes 20 through 2B.
- 20-2B These codes relate to the process of extending the current couple data set(s) of type *type*. The codes reflect the progress of the process. Determine the current couple data sets and complete the process manually. Depending on

the progress, you can determine the appropriate control card information either using the command INGPLEX CDS TYPE=*type* (reason > 24) or running the XCF utility against the temporary couple data set (reason > 21). If reason code 21 is issued, check the system log for an IXG257I or IXG261E message. If you find them, extend the couple data set(s) by increasing the DSEXTENT value by 1 or when not available by setting the value to 1. Otherwise and for reason code 20 determine the originator of the request to get the new item values.

Classes: 40, 43.

Module: INGRX305, INGRX711, INGRX720

AOF915I NO SPARE VOLUMES DEFINED FOR CDS TYPE *type*.

Explanation: The recovery of couple data sets of type *type* has been triggered. However, the automation could not complete the recovery because no spare volumes have been defined where a couple data set could be allocated.

System Action: Processing terminates.

Operator Response: Inform your system programmer.

System Programmer Response: Review the netlog and the system log. If the message has not been issued during the initialization of the automation, complete the recovery manually. To prevent any future occurrence of this message, modify the customization by adding one or more spare volumes for the named type of couple data sets. Then recycle NetView.

Classes: 40, 43.

Module: INGRX720

AOF916E NO MORE SPARE VOLUMES AVAILABLE FOR CDS type *type*.

Explanation: The recovery of couple data sets of type *type* has been triggered. However, the automation could not complete the recovery because all predefined spare volumes are either already in use, or they are unusable.

System Action: Processing terminates.

Operator Response: Inform your system programmer.

System Programmer Response: Review the netlog and the system log and complete the recovery manually. Then check the predefined spare volumes of the named type of couple data sets. When less than three volumes have been defined and the corresponding couple data sets are allocated to XCF add at least one more spare volume and then recycle NetView. Otherwise, determine the reason why the spare couple data sets or volumes could not be used. One reason can be that

there is not enough continuous space on the volume.

Classes: 40, 43.

Module: INGRX720

AOF917E THE HIGH-LEVEL QUALIFIER HAS NOT BEEN DEFINED.

Explanation: The recovery of couple data sets has been triggered. However, the automation could not complete the recovery because the high-level qualifier that was used to create a spare couple data set has not been defined.

System Action: Processing terminates.

Operator Response: Inform your system programmer.

System Programmer Response: Review the netlog and the system log and complete the recovery manually. To prevent any future occurrence of this message, disable the recovery for couple data sets. Or, modify the customization by defining the high-level qualifier and then recycle NetView.

Classes: 40, 43.

Module: INGRX720

AOF918I THE AUTOMATION IS GOING TO SWITCH THE COUPLE DATA SETS OF *type*

Explanation: The automation is going to switch the couple data sets of the specified type.

The variable *type* shows the type of the couple data sets that are being switched.

System Action: Processing continues.

Operator Response: None.

Classes: 40, 43.

Module: INGRX720

AOF919I *function* DETECTED STATUS *status* FOR CF *cfname*

Explanation: The function *function* detected a status change of the coupling facility inhibiting the function from further processing.

System Action: Processing terminates.

Operator Response: Press the REFRESH key and retry the function if possible.

System Programmer Response: None.

Classes: None.

Module: INGRX914

AOF920I NO SIGNALLING PATHS MATCH THE SPECIFIED CRITERIA *criteria*

Explanation: The program set up the commands MVS DISPLAY XCF,PATHIN,DEVICE=ALL and MVS DISPLAY XCF, PATHOUT,DEVICE=ALL. However, the message AOF920I was returned.

The variable *criteria* shows the path type (either PATHIN or PATHOUT).

System Action: Processing terminates.

Operator Response: None.

System Programmer Response: None.

AOF921I CONSOLE *console* IS INACTIVE

Explanation: The MVS command D R,LIST,T,CN=*console* was issued. However, no data was returned because the console is INACTIVE. Therefore, no requests can be displayed.

System Action: Processing terminates.

Operator Response: None.

System Programmer Response: None.

Classes: None.

Module: AOFRVX41

AOF923I NO ACTIVE TARGET SYSTEM FOUND ON TARGET HARDWARE *target* (LU=LUNAME)

Explanation: The operator issued the command GETSPCFP *luname*. Either the target hardware has not been initialized yet, or the LU name of the Support Element has changed.

System Action: Processing terminates.

Operator Response: Use the processor operations command ISQXDST to verify that at least one target system is initialized. If not, issue the command ISQXIII *target* where *target* is one of the target system names in the list. When the command completes successfully, retry the GETSPCFP command.

System Programmer Response: If the luname has changed, correct the name in the processor operations control file and restart processor operations.

Classes: 40 43.

AOF924A AUTOMATION OF MESSAGE IXC102A TERMINATED. REASON: *reason*[, {PROCESSOR OPERATIONS | BCPII} COMMAND RC=*rc*]

Explanation: XCF is removing a system from the sysplex and waits for the reply to proceed. The reply could not be automated.

The variable *reason* shows the reason code that was issued. It can have the following values:

- 10** The IXC102A message could not be formatted. The message identifier could not be located.
- 11** A command was issued to the MVS system which reported that a system left the sysplex. A timeout occurred while waiting for a reply to this command.
- 12** The proxy resource name for the system leaving the sysplex could not be determined.
- 14** The target system name of the ISQ900I message could not be obtained or the target system which sent the ISQ900I message is not initialized to processor operations.
- 15** The replyid for the IXC102A message could not be determined.
- 16** The system name could not be located in the IXC102A message.
- 17** The IXC102A message could not be retrieved for automation.
- 18** The automation requirements for the system leaving the sysplex could not be determined.
- 19** An error occurred while checking if the message was still outstanding.
- 30** A processor operations command failed. Refer to the appropriate command description.
- 31** A timeout occurred while waiting for the response of the Processor Management command.
- 32** The reply to the outstanding WTOR could not be sent.
- 33** An error occurred while determining the status of the local sysplex.

The variable *rc* shows the value of the return code.

System Action: Processing terminates.

Operator Response: Complete the shutdown of the system that left the sysplex. Then reply to the IEA102A message

System Programmer Response: Correct the problem. If reason code 11 was issued, no action is required.

Module: AOFRX700, AOFRX701, AOFRX702, INGRX705

Classes: 40 43

TEC: YES

**AOF925I AUTOMATION OF MESSAGE IXC102A
FOR *system* COMPLETED
SUCCESSFULLY**

Explanation: The automation of message IXC102A for system *system* omlpleted successfully. The system is no longer part of the sysplex.

System Action: None.

Operator Response: None.

System Programmer Response: None.

**AOF926I ERROR *error* DETECTED DURING
AUTOMATION OF MESSAGE IXC102A**

Explanation: The routine responsible for the automation of message IXC102A found a error.

- The variable *error* shows the error condition. The following error conditions can occur:

- | | |
|----|--|
| 10 | The message is not the IXC102A message. |
| 11 | The same message is being processed by another task. |
| 12 | Two or more commands were defined in the customization dialogs for message IXC102A. However, only one ISQCCMD can be issued. |
| 13 | The command defined for IXC102A message is not an ISQCCMD command. |
| 14 | The reply is no longer outstanding. |
| 15 | Incorrect call of a subsequent clist. |
| 30 | Using the supplied or default command, an attempt was made to deactivate the system leaving the sysplex but it failed. Another attempt will be made using the default command. |

System Action: Processing terminates for conditions 10, 11, 14, and 15. For condition 12, 13 and 30 processing continues using the default ISQCCMD.

Operator Response: None.

System Programmer Response: If error conditions 12 and 13 occurred, correct the definitions and reload the automation control file.

Module: AOFRX700, AOFRX701, AOFRX702, INGRX705

Classes: None.

TEC: NO

**AOF927E WTOR|WTOR BUFFER SHORTAGE
COULD NOT BE RESOLVED**

Explanation: The sysplex automation to recover from a WTO or WTOR buffer shortage did not complete successfully. One or multiple recovery stages signalled

an error. This is a transient status of the recovery. The message IEA406I/IEA232I for WTO/WTOR buffer shortage relief may indicate that the buffer shortage has been resolved.

System Action: Processing continues.

Operator Response: Check the system log for message IEA406I/IEA232I indicating a buffer shortage relief. If the message could not be found, perform manual recovery actions.

System Programmer Response: Search the system log for message IEE889I issued by the operating system internally. Identify the buffering consoles or jobs. If possible, requeue console buffers to the system log (K Q,L=*conname*). If possible, increase MLIM. Cancel jobs buffering messages or replies. After message IEA406I/IEA232I is issued, check if the current CONSOLxx definitions may need to be adjusted to avoid future buffer shortages.

Classes: 40, 43.

Module: INGRX730

**AOF928I SYSPLEX *sn bt* RECOVERY INCREASES
THE *in* VALUE TO *nv***

Explanation: The automation has detected a shortage of message or reply buffers. As a recovery result, the limit value is increased to avoid further buffer shortages. The changed value affects all active members in a sysplex, because the modified limit has sysplex scope.

The variable *sn* specifies the name of the sysplex.

The variable *bt* WTOR: write to operator with reply message buffers.

The variable *in* RMAX specifies the maximum buffer limit of write to operator with reply messages. If the system runs in SYSPLEX=LOCAL mode or the number of systems in the sysplex is greater than eight, the RMAX value is set to 9999. Otherwise it is not changed.

The variable *nv* shows a numeric number, representing the new limit value that is going to be active.

System Action: Processing continues.

Operator Response: Inform your system programmer.

System Programmer Response: This message may indicate that the current RMAX value of CONSOLxx member needs to be adjusted.

Classes: 40, 43.

Module: INGRX730

**AOF929I SYSPLEX *sn bt* RECOVERY SETS *st* ON
SYSTEM *sys***

Explanation: The automation has detected a shortage of message or reply buffers. As a recovery result, a limit value or a console setting is changed to avoid further buffer shortages. The changed value or console setting affect the named system in the sysplex.

The variable *sn* specifies the name of the sysplex.

The variable *bt* WTOR: write to operator with reply message buffers.

The variable *st* shows the string representing the command that was issued to recover from the buffer shortage. The limit values or console settings stay in effect until they are changed manually. In case of an extended recovery situation, this message may be issued multiple times showing commands increasing MLIM or RLIM values.

The variable *sys* specifies the name of the system.

System Action: Processing continues.

Operator Response: If this message is displayed frequently, inform your system programmer.

System Programmer Response: Multiple occurrence of this message may indicate that the CONSOLxx member specified limits or console definitions need to be adjusted.

Classes: 40, 43.

Module: INGRX730

**AOF930I HARDWARE STATUS CHANGE
DETECTED**

Explanation: The hardware monitoring program detected a status change of one or more hardware components defined during customization.

System Action: Processing continues.

Operator Response: None.

System Programmer Response: None.

Classes: None.

Module: INGRX051

**AOF931I VERIFICATION OF USER-DEFINED
CDS FAILED. RSN= *reason* DSN=*dsname***

Explanation:

The variable *reason* shows the value of the reason code. The following reason codes can occur:

- 1 The automation environment has not yet been initialized.
- 2 The data set name conflicts with the naming convention of the automation when creating a new couple data set.

4 The specified data set is not defined.

8 The specified data set is not available.

12 The specified data set is already allocated.

16 An unexpected error occurred.

20 The DYNALLOC macro failed.

The variable *dsname* shows the name of the data set.

System Action: Processing terminates.

Operator Response: Make sure that the name of the data set is correct.

System Programmer Response: Make sure that the data set is defined and available.

Classes: None.

Module: INGRX301

**AOF932I BOTH NAME AND VOLUME ARE
REQUIRED.**

Explanation: The user has only specified a user couple data set name or a volume.

System Action: Processing stops.

Operator Response: Specify both a data set name and a volume.

System Programmer Response: None.

Classes: None.

Module: INGRX301

AOF933I ACTION CODE IS NOT SUPPORTED.

Explanation: The action code is not supported for this entry for some program-related reason.

System Action: Processing stops.

Operator Response: See the panel help for the reason. Use a valid action code.

System Programmer Response: None.

Classes: None.

Module: INGRX300, INGRX904

AOF934I POLICY *policy* HAS BEEN STARTED.

Explanation: A 'SETXCF START,POLICY' command has been issued for the policy *policy*.

System Action: Processing continues.

Operator Response: Check that the policy becomes the active policy.

System Programmer Response: None.

Classes: None.

Module: INGRVX3A

AOF935I ENTER THE COUPLE DATA SET AND VOLUME.

Explanation: The automation is waiting for a user-defined couple data set to be specified.

System Action: None.

Operator Response: Enter the required information.

System Programmer Response: None.

Classes: None.

Module: INGRX301

AOF936E START OF POLICY *policy* FAILED.

Explanation: A "SETXCF START,POLICY" command has been issued for the policy *policy*. The command failed.

System Action: Processing stops.

Operator Response: Review the netlog.

System Programmer Response: None.

Classes: None.

Module: INGRX301

AOF937E SOME PENDING STRUCTURES CANNOT BE REBUILT AND REMAIN PENDING.

Explanation: The automation function responsible for releasing the 'POLICY CHANGE PENDING' status of the structures detected that one or more structures remain in this status.

System Action: Processing continues.

Operator Response: Use the command 'D XCF,STR,STATUS=POLICYCHANGE' to list the structures whose status is pending. Determine the reason why these structures remain in the status. Manually perform the necessary actions for releasing the status.

System Programmer Response: None.

Classes: None.

Module: INGRX919

TEC: NO

AOF938E DDNAME *ddname* IS PENDING.

Explanation: An automation function could not complete successfully because a required data set is not available.

The variable *ddname* shows the name of the data set that is not available.

System Action: Processing stops.

Operator Response: Notify your system programmer.

System Programmer Response: Add the ddname to the NetView startup procedure. To prevent a NetView recycle, allocate the data set dynamically using the ddname above.

Classes: None.

Module: INGRVX20, INGRX200

TEC: NO

AOF939I THERE IS NO ACTIVE POLICY.

Explanation: The automation detected that no policy is active at the moment.

System Action: Processing continues.

Operator Response: None.

System Programmer Response: None.

Classes: None.

Module: INGRX311, INGRX313, INGRX314

AOF940I *resource(s)* AUTOMATION IS/ARE NOT ENABLED.

Explanation: The selected function is not enabled for automation.

System Action: Processing terminates.

Operator Response: Enable the function by setting the RECOVERY flag of one of the named resources to 'Y'. Then, retry the function.

System Programmer Response: If the function should be enabled by default, update the customization values.

Classes: None.

AOF941I CFRM COUPLE DATA SET IS NOT AVAILABLE.

Explanation: The CFRM active policy could not be read because the couple data set supporting TYPE(CFRM) is not accessible from this system.

System Action: Processing terminates.

Operator Response: Issue the request from a system with access to a CFRM couple data set.

System Programmer Response: None.

Classes: None.

Module: INGRVX9Q

AOF942I NO POLICIES DEFINED.

Explanation: There are no policies defined for the selected couple data set type.

System Action: None.

Operator Response: Inform your system programmer.

System Programmer Response: Define policies as necessary.

Classes: None.

Module: INGRX311, INGRX313, INGRX314

AOF943I NO LOG STREAMS OR STRUCTURES DEFINED.

Explanation: There are no log streams or structures defined in the LOGR couple data set.

System Action: None.

Operator Response: Inform your system programmer.

System Programmer Response: Define log streams and structures as necessary.

Classes: None.

Module: INGRX312

AOF944I CDS TYPE *cdstype* IS NOT ENABLED ON SYSTEM *system*.

Explanation: The couple data set type cannot be displayed because it is not enabled on the target system.

System Action: None.

Operator Response: None.

System Programmer Response: None.

Classes: None.

Module: INGRX310

AOF945I CDS TYPE *cdstype* IS NOT ACCESSIBLE ON SYSTEM *system*.

Explanation: The couple data set type cannot be displayed because it is not accessible on the target system.

System Action: None.

Operator Response: None.

System Programmer Response: None.

Classes: None.

Module: INGRX310

AOF946I SOME SENDER PATH INFORMATION COULD NOT BE DETERMINED FOR *cfname*.

Explanation: Currently not all systems in the sysplex have access to the coupling facility *cfname*. However, the automation cannot determine the sender paths of some or all of those systems.

System Action: Processing continues.

Operator Response: If the local system does not show the control unit of the coupling facility, try the command from a system that shows the information when you enter the MVS command 'D CF,CFNM=*cfname*'.

System Programmer Response: None.

Classes: None.

Module: INGRVX9P

AOF947I *action* OF STRUCTURE *structure* IS NOT SUPPORTED.

Explanation: The automation detected that the action selected for the structure is either not supported or currently unavailable.

The variable *action* shows the action to be performed.

The variable *structure* shows the name of the structure.

In case of SYSTEM-MANAGED actions the cause of the problem could be that the CFRM couple data set is not formatted properly.

System Action: The action is not performed.

Operator Response: None.

System Programmer Response: None.

Classes: None.

Module: INGRX904

Classes: None.

TEC: NO

AOF948I FUNCTION NO LONGER SUPPORTED.

Explanation: Due to some changes in a new version of this product this function has either been replaced or deleted.

System Action: Processing terminates.

Operator Response: If the function was replaced, use the new function.

System Programmer Response: None.

Classes: None.

Module: INGRVX90

AOF949I NO DUMP DATA SET AVAILABLE FOR JOB *jobname*

Explanation: The automation tries to take a dump before the job is cancelled. Neither system dump data sets nor automatic allocated dump data sets were available.

System Action: Processing continues.

Operator Response: Inform your system programmer.

System Programmer Response: To fix the error do one of the following:

- Add more system dump data sets or volumes for automatic allocated dump data sets
- Clear full dump data sets to prevent further occurrences of this message.

Module: INGRX741

Classes: 40

TEC: NO

AOF950I **JOB *jobname* HOLDS AN ENQ FOR *nnn* SECONDS ON RESOURCE *resource***

Explanation: The automation found a long blocking ENQ on the resource by the specified job. The policy defined during customization does not allow the cancellation of the job to free the resource.

 The variable *jobname* shows the name of the job.

 The variable *nnn* shows the waiting time in seconds. A value of 999 is the maximum waiting time shown.

 The variable *resource* shows the name of the resource where the ENQ was found.

System Action: Processing continues.

Operator Response: Inform your system programmer.

System Programmer Response: If the waiting time is feasible, remove the resource entry from the policy or increase the time if applicable. Otherwise allow the cancellation of the job if necessary.

Module: INGRX741

Classes: 40 43

TEC: NO

AOF951I **DUE TO A LONG ENQ DETECTION THE JOB *jobname* IS BEING CANCELLED.**

Explanation: The automation detected a long blocking ENQ held by the specified job. The job is being cancelled based on policy definitions.

 The variable *jobname* shows the name of the job.

System Action: The job is cancelled with the MVS CANCEL command.

Operator Response: None.

System Programmer Response: None.

Module: INGRX741

Classes: 40

TEC: NO

AOF952I **DUE TO A AUXILIARY STORAGE SHORTAGE DETECTION THE JOB *jobname* IS BEING CANCELLED**

Explanation: The automation found an auxiliary storage shortage. The system issued a message indicating the job had rapidly increasing storage requests. The job is being cancelled based on policy definitions.

 The variable *jobname* shows the name of the job.

System Action: The job is cancelled with the MVS CANCEL command.

Operator Response: None.

System Programmer Response: None.

Module: INGRX742

Classes: 40

TEC: NO

AOF953E **RECOVERY OF AUXILIARY STORAGE SHORTAGE COULD NOT COMPLETE SUCCESSFULLY. REASON=*reason***

Explanation: The automation could not recover an auxiliary storage shortage.

 The variable *reason* shows the reason code. It can have the following values:

Reason	Description
4	The recovery was enabled, but neither spare volumes for dynamically allocated page data sets nor preallocated page data sets were defined.
8	The spare volumes for dynamically allocated page data sets and the predefined local page data sets are used up.
12	The spare volumes for dynamically allocated page data sets and the predefined local page data sets are used up. Some of them have been marked unusable because the system rejected the data sets when the automation tried to make them available.
16	The IDCAMS service failed for the dynamic allocation of a page data set. Check the netlog for details.
20	The PAGTOTL value defined in IEASYSxx member used during IPL does not allow to add more local page data sets.
24	The system issued a message indicating a problem when the automation tried to make a local page data set available.

System Action: Processing terminates.

Operator Response: Inform your system programmer.

System Programmer Response: If reason codes 4, 8,

12, or 16 occurred, add more local page data sets manually. If reason code 20 occurred, follow the instructions given in messages IRA200E and IRA203I. For reason code 24 follow the instructions of the messages IEE78n issued prior to this message.

Module: INGRX742

Classes: 40 43

TEC: NO

AOF954I PAGE DATA SET *dsname* SUCCESSFULLY ADDED.

Explanation: Due to a recovery situation the automation successfully added a local page data set to the system.

The variable *dsname* shows the name of the data set.

System Action: Processing continues.

Operator Response: None.

System Programmer Response: Delete the page data set via the PAGEDEL command when the data set is no longer needed. The data set is then returned to the list of spare page data sets being used by the automation. If the data set was dynamically allocated you can also delete it physically when no other page data set is active on the volume where the data set is allocated.

Module: INGRX742

Classes: 40 43.

TEC: NO

AOF955I NO DIFFERENCES FOUND

Explanation: A comparison of IPL information did not indicate any differences.

System Action: Processing continues.

Operator Response: None.

System Programmer Response: None.

Module: INGRX200, INGRX201

Classes: None.

TEC: NO

AOF956I IPL INFORMATION OF *timestamp* FROM *sysname* HAS BEEN DELETED

Explanation: All IPL information collected for the system after it was IPLed at *timestamp* was deleted.

The variable *timestamp* shows the time of the IPL.

The variable *sysname* shows the name of the system.

System Action: Processing continues.

Operator Response: None.

System Programmer Response: None.

Module: INGRVX20

Classes: None.

TEC: NO

AOF957I LOG ENTRY: *entry*

Explanation: The automation has logged an intermediate result of an automated process.

The variable *entry* shows the log entry.

System Action: Processing continues.

Operator Response: None.

System Programmer Response: None.

Module: INGRVX92

Classes: None.

TEC: NO

AOF958I DUMP COULD NOT BE TAKEN. RC=*rc* RSN=*rsn*.

Explanation: The requested dump could not be taken.

The variable *rc* shows the return code. It can have the following values:

RC	Description
4	Only partial dump was taken.
8	SDUMPX macro error occurred.
12	An internal error occurred.
16	An internal error occurred.
20	An internal error occurred.
24	No private storage is available.
28	No common storage is available.

The variable *rsn* shows the reason code.

System Action: Processing continues.

Operator Response: Depending on the return code do the following:

RC	Description
4	Search the system log for message IEA911E for the reason of the partial dump.
8	Refer to the return and reason code description of the SDUMPX macro for the reason why the dump could not be taken.
12, 16, 20	Contact your IBM service representative.
24, 28	Retry the function later.

System Programmer Response: None.
Module: INGRX262
Classes: None.
TEC: NO

AOF959I SYSTEM *sysname* IS NOT REGISTERED.

Explanation: The requested system is not registered to the automation. However, this function requires the registration because it invokes system services locally.

System Action: Processing continues.

Operator Response: Override the system name with blanks to force the selection panel showing the system being registered. Or, start the automation on the requested system using the same XCF group as the local system.

System Programmer Response: None.

Module: INGRX260

Classes: None.

TEC: NO

HSAH001I HSAET32 SERVICES *verb* COMPLETE.

Explanation: Processing of the command completed successfully.

The variable *verb* shows the name of the HSAET32 command verb that completed successfully.

System Action: HSAET service processing continues as requested by the command verb.

Operator Response: None.

System Programmer Response: None.

HSAH002I INVALID HSAET32 SERVICES REQUEST *request*.

Explanation: The identified HSAET32 command is not valid.

The variable *request* shows the name of the HSAET32 command that is not valid.

System Action: None.

Operator Response: Correct and reissue the command as necessary. Valid services are REBUILD,START,STATUS,DISABLE,ENABLE.

System Programmer Response: None.

HSAH003I HSAET32 SERVICES REQUEST FAILED, RC=*rc*, REASON *rsn*.

Explanation: An error occurred during an attempt to process a valid HSAET32 command.

The variable *rc* shows the return code that was issued.

The variable *rsn* shows the reason code that was issued.

System Action: None.

Operator Response: Issue an HSAET32 STATUS command and verify that the status reported by message HSAH004I is as expected. Then reissue the command as appropriate. If the problem persists, notify the system programmer.

System Programmer Response: This message appears with return code 8 and reason code 20 (14x) as a result of an HSAET32 START command if ET32 services were started previously (usually done during NetView initialization). If other problems occurred, contact your IBM support center.

HSAH004I HSAET32 SERVICES ARE *status*.

Explanation:

The variable *status* shows the current status of ET32 services. The following values can occur:

INACTIVE

HSAET32 services were not initialized and are unavailable.

ENABLED

HSAET32 services were initialized and are available.

DISABLED

HSAET32 services were initialized but were disabled from use.

System Action: None.

Operator Response: Verify that the status indicated is as currently required.

System Programmer Response: None.

HSAM5160I VSAM *request* ERROR OCCURRED. RC=*rc* REASON=*reason*

Explanation: The VSAM macro instruction that corresponds to the request failed.

The variable *request* shows the request that failed.

The variable *rc* shows the return code that was issued.

The variable *reason* shows the reason code that was issued.

System Action: Processing terminates.

Operator Response: Inform your system programmer.

System Programmer Response: Check the return and reason codes in *DFSMS/MVS Macro Instructions for Data Sets*, in the Chapter *VSAM Macro Return and Reason Codes*. Correct the error accordingly. Then rerun the program with the PARM parameter 'RERUN'.

Module: HSAPSIPL

Classes: None.

TEC: NO

**ING800I VERIFYING *type-model* HARDWARE
AUTOMATION INTERFACE**

Explanation: A check of the type and model of the hardware automation interface is performed.

The variable *type-model* shows the type and model of the hardware being verified.

System Action: The verification process for the specified processor is starting.

Operator Response: None.

System Programmer Response: None.

**ING801I AUTOMATION INTERFACE NOT
SUPPORTING *type-model* HARDWARE**

Explanation: The verification determined that the processor hardware is not supported

The variable *type-model* shows the type and model of the hardware.

System Action: None.

Operator Response: None.

System Programmer Response: None.

**ING802I HARDWARE INTERFACE
COMPONENT HSAET32 CANNOT BE
ACTIVATED**

Explanation: An error was reported by the HSAET32 internal command. The command that establishes and activates the hardware interface failed.

System Action: None.

Operator Response: None.

System Programmer Response: Check the netlog or the system log for HSAxxxx messages about the problem.

**ING803I HARDWARE INTERFACE AUTO
OPERATOR *autotask* CANNOT BE
STARTED**

Explanation: The named autotask failed to start. The hardware interface cannot be used.

The variable *autotask* shows the autotask that could not be started.

System Action: None.

Operator Response: None.

System Programmer Response: Check the netlog for additional messages. Correct the problem and activate

the autotask again. This will initialize the hardware interface.

**ING804I HARDWARE AUTOMATION
INTERFACE OF *type-model* PROCESSOR
VERIFIED**

Explanation: The verification for the specified processor has ended.

The variable *type-model* shows the type and model of the hardware.

System Action: None.

Operator Response: None.

System Programmer Response: None.

ING805I CONTACTING CPC *cpcaddr*

Explanation: The hardware interface tries to contact the processor hardware.

The variable *cpcaddr* shows the address of the CPC.

System Action: None.

Operator Response: None.

System Programmer Response: None.

**ING806I COMMUNICATION WITH CPC *cpcaddr*
CANNOT BE ESTABLISHED**

Explanation: A failure occurred while the processor hardware was contacted through the hardware interface.

The variable *cpcaddr* shows the address of the CPC with which no session could not be established.

System Action: None.

Operator Response: None.

System Programmer Response: Check the netlog for AOFAXxxx messages to obtain more information, for example the available condition and sense codes. Correct the problem and restart the session to the PC.

**ING807I HARDWARE COMMUNICATION
MODULE *modname* NOT ACTIVE**

Explanation: The hardware interface communication module was found not to be active. The hardware interface is not active.

The variable *modname* shows the name of the module which is not active.

System Action: None.

Operator Response: None.

System Programmer Response: Check the netlog for additional messages. Verify that the autotask name defined for autooperator HWOPER02 has operator

profile AOFPRFHW defined. Correct the problem and restart the autooperator.

ING808I **SYSTEM *sysname* IMAGE STATUS ON**
 cpcaddr **IS: *istat***

Explanation: The hardware interface reports the status of the specified z/OS system.

The variable *sysname* shows the name of the system for which the image status was reported.

The variable *cpcaddr* shows the address of the CPC.

The variable *istat* shows the image status. The following values can occur:

- OPERATING
- NOT_OPERATING
- NOT_ACTIVATED
- STATUS_CHECK
- EXCEPTIONS
- POWERSAVE

System Action: None.

Operator Response: None.

System Programmer Response: None.

ING809I **HARDWARE STATUS OF SYSTEM**
 sysname **CANNOT BE DETERMINED**

Explanation: An error occurred when retrieving the image status for the specified system.

The variable *sysname* shows the name of the system of which the hardware status could not be determined.

System Action: None.

Operator Response: None.

System Programmer Response: Check the netlog for AOFAXxxx messages to obtain more information, for example, the available condition and sense codes. Use an HMC to determine if the image (LPAR) that this system is configured for, is defined and activated.

Part 7. Appendices

Appendix A. Making security definitions using the command authorization table

Defining security using msys for Operations definitions

Use the information in this Appendix to define your security using only msys for Operations definitions instead of using a SAF product. msys for Operations delivers a CNMSCAT1 sample file. The sample predefines security settings for your use. To use the sample, copy the CNMSCAT1 file from NETVIEW.VnRnMn.CNMSAMP to your DSIPARM user domain. The following assumes that you either tailor the file to your enterprise's requirements, or that you issue the necessary commands from the system console.

Note, that the HSAET32 BCP interface requires the use of a SAF product. If you plan to use the CF ENABLE, CF DRAIN, and IXC102A automation functions, it is recommended to use a SAF related OPERSEC value. MINIMAL or NETVPW is not recommended.

Overview of operator security

Operator security can be defined using msys for Operations, an SAF product such as RACF, or a combination of both. The OPERSEC keyword in the OPTIONS statement determines which type of operator security is used. Use the REFRESH command to change the value of the OPERSEC keyword.

Review the information in Table 13 to determine which type of operator security best meets your requirements.

Note: It is recommended to use an SAF product exclusively (OPERSEC=SAFDEF) for operator security.

Table 13. Operator security definition types

OPERSEC value	Type of operator password and logon attributes
MINIMAL	Both operator passwords and logon attributes are ignored.
NETVPW	Operator passwords and logon attributes are provided exclusively by msys for Operations. Operator passwords are specified in DSIOPF. Logon attributes are specified in member DSIPRF and defined in DSIPRF.
SAFDEF	Operator passwords and logon attributes are provided exclusively by an SAF product. Operator passwords are checked by an SAF product, and logon attributes are defined in the NetView segment of an SAF product. Access to the data sets protected in the DATASET class and to MVS system commands protected in the OPERCMDS class of the SAF product are checked at the individual task level.
SAFCHECK	Operator passwords and logon attributes are provided by a combination of msys for Operations and an SAF product. Operator passwords are checked by an SAF product, with operator profiles specified in msys for Operations member DSIOPF and logon attribute values defined in DSIPRF. Access to the data sets protected in the DATASET class and to MVS system commands protected in the OPERCMDS class of the SAF product are checked at the individual task level.

Defining security using msys for Operations definitions

For a description of the OPERSEC keyword, see Chapter 16, “msys for Operations Definition Statements Reference” on page 159.

By defining msys for Operations operators exclusively to an SAF product, you eliminate the need for members DSIOPF and DSIPRF. See “Defining operators, passwords, and logon attributes” on page 49 for more information on defining operators and operator attributes using an SAF product.

Operator identifiers

Define a unique operator identifier for each operator who logs on to msys for Operations.

Do not use the names of msys for Operations commands, components, printers (hardcopy logs), terminals, or task identifiers for operator identifiers. Also, do not use the following reserved keywords:

ALL	NNT
DPR	OPT
DST	OST
HCL	PPT
HCT	SYSOP
LOG	TCT
MNT	

Using msys for Operations for password authorization

Use password security to prevent unauthorized logging on to msys for Operations. To use msys for Operations for password authorization, specify OPERSEC=NETVPW on the OPTIONS statement in DSIDMN or on the msys for Operations REFRESH command. The password stored in DSIOPF is used to check logon password authorization, so you must update DSIOPF to change a password.

Note: To prevent unauthorized viewing or modification of DSIOPF and command lists which contain passwords, see “Controlling access to data sets and members” on page 213.

Define the operator identifier and password with the OPERATOR definition statement in DSIOPF as follows:

```
NEWOPER  OPERATOR  PASSWORD=NEWOPER
          PROFILEN  DSIPROFB
```

Where NEWOPER is the operator identifier and NEWOPER is the operator password.

Using msys for Operations without password authorization

When OPERSEC=MINIMAL is defined in the OPTIONS statement, msys for Operations does not perform any password checking. Unless you use other ways of keeping your system secure, such as physically restricting access to terminals, you should use password security.

Operator logon attributes

Operator logon attributes describe characteristics associated with an operator.

Operator logon attributes can be defined in msys for Operations, in an SAF product, or in both. Although only one definition can be in effect at a time, you

Defining security using msys for Operations definitions

can dynamically change whether operator logon attributes are used from msys for Operations operator profiles (DSIPRF) or the NetView segment of an SAF product.

Whether you define operator profiles in DSIPRF or define operators in an SAF product, altering the logon attributes will not have an effect on the task until it is logged off, then logged on again. Before altering or migrating operator definitions, you should understand the following operator attributes:

- IC keyword on a PROFILE statement
- MSGRECVR keyword on an AUTH statement

Using MSGRECVR

The MSGRECVR attribute of the AUTH statement can be used in both msys for Operations operator profiles and in the NetView segment of an SAF product. It specifies whether operators are eligible to receive unsolicited messages that are not routed to a particular operator using either the msys for Operations ASSIGN command or msys for Operations automation.

For more information about using the MSGRECVR keyword in a DSIPRF profile, see to “AUTH” on page 160.

Using IC

If the IC keyword is specified, it must be the last keyword on the PROFILE statement. For more information about using the IC keyword in a DSIPRF profile, see “PROFILE” on page 169.

Defining operator attributes in msys for Operations profiles

You can code more than one profile for an operator. You can also use the same profile for more than one operator. For each operator profile, create a profile member in DSIPRF with a PROFILE definition as the first statement in that file. Other definition statements, such as AUTH, follow this PROFILE statement.

Here is an example showing how you could add an operator definition to the DSIOPF member of DSIPARM:

```
NEWOPER  OPERATOR  PASSWORD=NEWOPER
          PROFILEN  DSIPROFA
```

You can define profiles that:

- Specify a command or a command list to run automatically when an operator logs on
- Specify whether an operator is eligible to be the authorized receiver of undeliverable messages.

For examples of profile definitions, browse profiles DSIPROFA and DSIPROFB.

Here is an excerpt of sample profile DSIPROFA:

```
DSIPROFA  PROFILE  IC=LOGPROF1
          AUTH      MSGRECVR=NO,CTL=GLOBAL
          END
```

Note: msys for Operations does not use the CTL statement defined in DSIPROFA.

The profile in the previous example specifies:

IC=LOGPROF1

A command list named LOGPROF1 (CNME1049) is run automatically when an operator logs on with this profile.

Defining security using msys for Operations definitions

MSGRECVR=NO

Operator is not eligible to be the authorized receiver.

You can define other profiles as necessary by creating additional profile members in DSIPRF. For more information about creating profile members, see “PROFILE” on page 169.

Dynamically adding or deleting operators

If you are using msys for Operations operator definitions in DSIOPF, you can use the msys for Operations REFRESH OPERS command to dynamically add or delete operators while msys for Operations is running. This command refreshes operator definitions in DSIOPF that were added since the last time msys for Operations was stopped and restarted or since the last REFRESH OPERS command was issued. You can also use the REFRESH OPERS,TEST command to check the operator definitions that will change when you issue the REFRESH OPERS command.

Command authorization

This section provides the following information about command authorization:

- An overview of command authorization
- Command authorization using the command authorization table
- Command authorization for specific commands

It is recommended that you review this section in its entirety before you begin to define your command security.

Overview

Command authorization is the process of protecting commands from unauthorized use.

A *command authorization table* (CAT) enables you to restrict access to commands, keywords, and values. It then allows you to permit operators and groups of operators to access these restricted commands, keywords, and values. You can also specify commands, keywords, and values that pass authorization checking. The NVSS command authorization table is stored as a member of DSIPARM. You can use the REFRESH command to dynamically update your table. See “Command authorization using a command authorization table (CAT)” on page 196 for more information.

Recommended commands to protect: Customers must decide which commands to protect based on their unique security requirements. However, it is recommended that, at a minimum, you restrict the following commands, because they can affect the msys for Operations environment or access to it:

- AFTER (Use of the PPT keyword.)
- AT (Use of the PPT keyword.)
- AUTOTBL
- CHRON (Use of the ROUTE keyword. See “Defining security for the CHRON command” on page 209 for more information.)
- CLOSE
- DEFAULTS
- EVERY (Use of the PPT keyword)
- EXCMD. (See the msys for Operations online help for more information.)
- FOCALPT
- GETCONID
- GLOBALV
- INGRCCCHK

- MODIFY
- MVS (See “MVS” on page 100 for more information.)
- OVERRIDE
- PURGE
- READSEC (See the online help for more information.)
- REFRESH
- RMTCMD
- RUNCMD
- SUBMIT (See the online help for more information.)
- SETCONID
- START
- STOP
- VARY
- WRITESEC (See the online help for more information.)

Exceptions to command authorization checking: Major exceptions to command authorization checking include:

- Commands entered as replies to the msys for Operations WTOR (message DSI802A) are not authority checked. To prevent users from issuing commands using the WTOR, specify CMDWTOR=NO in the MVSPARM statement in DSIDMN. This prevents msys for Operations from issuing the WTOR.
- Command authority checks are not made against the PPT or DST tasks; therefore, you do not need to authorize these tasks to access your protected commands.
- Commands issued from a source ID of *BYPASS* are not checked for command authorization by The NVSS command authorization table. The SOURCEID will default to *BYPASS* if the command was entered at an extended multiple console support (EMCS) console and the operator was not logged on to the EMCS console.

Auditing command authority checking: You can audit accesses to protected commands, keywords, and values. This auditing can be done on an individual command, keyword, or value basis.

For command authorization using a NVSS command authorization table, you can specify the CATAUDIT keyword on the DEFAULTS command to determine the level of auditing performed. With the option not to audit, you can chose to audit unsuccessful or failed attempts to access protected command identifiers, or to audit all matches on command identifiers in your table. For more information on the DEFAULTS command, refer to the NVSS online help. You can also specify specific auditing levels on specific command identifiers using PROTECT and EXEMPT statements with the AUDIT keyword, as described in “Table Statements” on page 198.

If auditing is specified, the records are written to SMF as record type 38.

Protecting commands containing special characters: There are some special characters that cannot be included in the command identifier or SAF resource name. For this reason, NVSS translates these special characters to other characters before passing them to either the NVSS command authorization table. The special characters that are translated along with their translated results are:

Reserved Character	Translated Result
.	/
*	+
%	?

Command authorization

&	:
- (dash)	_ (underscore)
' ' (blank)	_ (underscore)

As an example, the following NVSS command can be entered by a NVSS operator:
LIST MEMSTAT=*

To restrict access to the FPCAT keyword and its value in the command authorization table, include the following statement:

```
PROTECT NETA.MS001.LIST.MEMSTAT.+
```

Note that the asterisk was translated to a plus.

Command authorization using a command authorization table (CAT)

NVSS provides the ability to use a NVSS command authorization table to restrict the use of commands and operands to specific operators or groups of operators. The table consists of a member in DSIPARM containing the authorization statements. This table can include statements to embed other members from DSIPARM. Using a NVSS command authorization table, you can also protect command lists that do not have a CMDMDL statement in DSICMSYS. The sample command authorization table supplied with msys for Operations is CNMSCAT1. The following sections provide the information that you need to modify CNMSCAT1 to meet your security requirements.

Command identifiers: Identify which commands, keywords, and values are protected using command identifiers. The format of the command identifiers for the NVSS command authorization table is the same as the format of the resource names used in the NETCMDS class in an SAF product. In its full form, a command identifier uses these fields: *netid.luname.command.keyword.value*.

Use the NVSS LISTVAR command to determine the *netid* and *luname* values for your systems. In the LISTVAR example shown in Figure 51, the current values are NETA for *netid* and the NVSS domain, MSO01, for *luname*.

```
LISTVAR
CNM353I LISTVAR : OPSYSTEM = MVS/ESA
CNM353I LISTVAR : MVSLEVEL = SP5.1.0
CNM353I LISTVAR : CURSYS   = VTAM430
CNM353I LISTVAR : VTAMLVL  = VT43
CNM353I LISTVAR : VTCOMPID = 5695-11701-301
CNM353I LISTVAR : NETVIEW  = NV31
CNM353I LISTVAR : NETID   = NETA
CNM353I LISTVAR : DOMAIN  = MSO01
CNM353I LISTVAR : APPLID   = MS001007
CNM353I LISTVAR : OPID     = OPER3
CNM353I LISTVAR : LU       = A01A703
CNM353I LISTVAR : TASK     = OST
CNM353I LISTVAR : NCCFCNT  = 0
CNM353I LISTVAR : HCCOPY   =
CNM353I LISTVAR : CURCONID =
CNM353I LISTVAR : DATE     = 11/03/94
CNM353I LISTVAR : TIME     = 13:41
```

Figure 51. Example of LISTVAR command output

Note that some characters are reserved if you are using the NVSS command authorization table or an SAF security product for command authorization checking.

The command identifier can be up to 246 characters in length, including the periods that serve as field delimiters. The individual fields of the command identifier have no maximum length as long as the entire command identifier length does not exceed 246 characters.

You can use generic characters in command identifiers. An asterisk (*) can be used to indicate that all possible values of a field are protected or permitted, except those that are more explicitly specified. You can use the asterisk either as a replacement for a field or as a trailing character to indicate that all items that begin with the specified characters are to be protected. The percent sign (%) can be used as a single character generic anywhere within the command identifier. Generic characters are useful to specify a level of protection for commands for which there is not a match in the table. You can do this because the most specific command identifier determines the level of protection for a command.

Note: The generic character combination %* is not valid.

Commands are checked separately from keywords and values. When designing command identifiers, keep in mind that the command is checked first, in addition to the subsequent security checking for the command, keyword, and value combinations. Keywords and their associated values are checked as a pair. To protect a keyword that has a value associated with it, there must be an entry in the value position of the command identifier. The command identifiers can be in these formats:

netid.luname.command

netid.luname.command.keyword (used only for keywords without values)

netid.luname.command.keyword.value

Where:

netid

Indicates the VTAM network identifier. You can specify a generic character (*) for this field.

The *netid* specification is syntax checked for format (*netid* may not begin with a left parenthesis) but no checking is done to verify that the *netid* specified matches the current *netid*. This field is treated as a place holder and is supported so that the format of the command identifier in the NVSS command authorization table is the same as the format of a resource name in the NETCMDS class of an SAF product.

luname

Indicates the domain identifier. Only statements which match your *luname* are loaded when the NVSS command authorization table is activated, but all statements are syntax checked, regardless of *luname*.

command

Indicates the command name on the CMDMDL statement in the DSICMSYS member of DSIPARM, or a command list name. This must be the actual command name and not a synonym defined by the CMDSYN statement. No checking is done to validate that *command* is a valid command or command list name.

keyword

Indicates the keyword identifier that is protected.

value

Indicates the value identifier that is protected when used with the keyword on the command.

Command authorization

Command authorization table syntax: Table statements consist of free-form text which specify a table statement type followed by its operands. You can enter the text in upper or lowercase, with the exception of %INCLUDE statements, which must be in uppercase. For all other statements, the text is converted to uppercase when the table is processed. The table statements must be coded between columns 1 and 72. If a statement is too long to fit between columns 1 and 72, you can use the <BEGIN> and <END> statements when multiple lines should be treated as a single statement. You can include a sequence number in columns 73 through 80 for problem determination purposes. If NVSS encounters any errors while processing the table statements, the error messages issued include the sequence number of the line in error. An asterisk in column 1 denotes a comment and causes the rest of the NVSS command authorization table line to be ignored.

Table Statements: This section describes the format and function of the following statement types:

- <BEGIN> and <END>
- %INCLUDE
- PROTECT
- EXEMPT
- GROUP
- PERMIT
- SETVAR

<BEGIN> and <END> Statements: The <BEGIN> and <END> statements specify the beginning and end of a NVSS command authorization table statement that spans multiple input lines. The total length of any individual table statement must not exceed 4096 characters, including blanks, which provides a maximum of 56 input lines.

The syntax for the <BEGIN> and <END> statements are:

<BEGIN>

▶▶<BEGIN>—————▶▶

<END>

▶▶<END>—————▶▶

The <BEGIN> and <END> statements must appear on lines by themselves. Command identifiers may continue onto more than one line, but statement types, group names, and each user ID in a userid_list should not span more than one line. As you enter multiple input lines, be careful not to accidentally put an asterisk in column 1, because the remainder of that line will be treated as a comment.

Example:

To enter a command authorization statement that spans two input lines, use the following:

```
<BEGIN>
GROUP ALLOPS OPER1,OPER2,OPER3,OPER4,OPER5,OPER6,OPER7,OPER8,OPER9,
          OPER10,OPER11
<END>
```

Note: Blank characters between input lines alignment are valid.

%INCLUDE statement: The %INCLUDE statement enables you to keep portions of your NVSS command authorization table in separate DSIPARM members. Both the %INCLUDE statement and its values (either the *membername* or *&varname*) must be capitalized.

The syntax for the %INCLUDE statement is:

%INCLUDE statement



Where:

```
%INCLUDE
```

Indicates the keyword coded at the beginning of each %INCLUDE statement.

membername

Indicates the name of the DSIPARM member to be included.

 $\&varname$

Indicates the name of an existing local or global variable, preceded by the ampersand (&) character.

Usage notes:

1. Each %INCLUDE statement can be no longer than one line.
2. A member that has been included can contain %INCLUDE statements as well as other NVSS command authorization table statements.
3. A member that has been included cannot include itself either directly or indirectly.
4. If you specify a variable name for the value of the %INCLUDE, the NVSS program includes the designated member when you issue the REFRESH command with CMDAUTH=TABLE. You cannot use a variable name in a command authorization table specified on an OPTIONS statement in DSIDMN for NVSS initialization. NVSS searches for the variables in the following order:
 - If the REFRESH command is issued from a command procedure, the NVSS program searches first for a local variable of the name *varname*, then for a task global variable, and finally for a common global variable.
 - If the REFRESH command is not issued from a command procedure, the NVSS program searches for a task global variable of the name *varname* and then for a common global variable.

If you change the value of the variable after activating the NVSS command authorization table, the member that is included does not change, unless you reissue the REFRESH command.

Example:

To include member TBL02 from DSIPARM, include the following statement in your NVSS command authorization table:

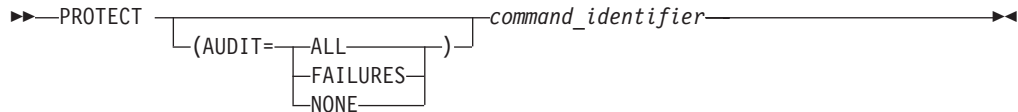
```
%INCLUDE TBL02
```

Command authorization

PROTECT statement: The PROTECT statement identifies a command identifier to be protected.

The syntax for the PROTECT statement is:

PROTECT



Where:

command_identifier

Specifies the *netid*, *luname*, command, keyword, and value to be protected. See “Command identifiers” on page 196 for information on specifying command identifiers.

AUDIT

Specifies whether an audit record should be created when a command authority check yields a match on the command identifier. The audit records can be SMF type 38 records, or the DSIXITXL exit can write the records to an external log. The AUDIT keyword is optional. If not specified, auditing is determined by the value of CATAUDIT on the DEFAULTS command. When specified, the value overrides the value specified for CATAUDIT on the DEFAULTS command. Valid values for AUDIT are:

ALL

Specifies that an audit record is to be created when a match occurs on the command identifier.

FAILURES

Specifies that an audit record is to be created when a match occurs on the command identifier and the command authority decision is *fail*.

NONE

Specifies that no audit record is to be created when a match occurs on the command identifier.

Example:

To define a command identifier to protect the AUTH keyword and MASTER value of the GETCONID command in domain MSO01, use the following statement:

```
PROTECT *.MSO01.GETCONID.AUTH.MASTER
```

To define a command identifier to protect the AUTH keyword and MASTER value of the GETCONID command in domain MSO01, and to create audit records for all attempts to get a console with master authority, use the following statement:

```
PROTECT (AUDIT=ALL) *.MSO01.GETCONID.AUTH.MASTER
```

Usage notes:

- Create one PROTECT statement for each command that you want to protect. For example, to protect the STOP command for *luname* MSO01, create a table entry as follows:

```
PROTECT *.MSO01.STOP
```

- Create one PROTECT statement for each command and keyword that does not have an associated value which you want to protect. For example, to protect the OFF keyword on the AUTOTBL command for *luname* MSO01, create a table entry as follows:

```
PROTECT *.MSO01.AUTOTBL.OFF
```

- Create one PROTECT statement for each command, keyword, and value combination that you want to protect. For example, to protect the FORCE keyword with a value of AUTO1 on the STOP command for *luname* MSO01, create a table entry as follows:

```
PROTECT *.MSO01.STOP.FORCE.AUTO1
```

- To protect all values of FORCE in the previous example, create a table entry as follows:

```
PROTECT *.MSO01.STOP.FORCE.*
```

- To protect all values of FORCE that begin with "TEST" and end with "0", create a table entry as follows:

```
PROTECT *.MSO01.STOP.FORCE.TEST%0
```

- To allow an NVSS operator to issue an NVSS command that is protected with a PROTECT statement, you must use a PERMIT statement for each operator ID or group of operators that should be authorized.

- If you have more than one statement that describes the same command, keyword, and value, the first is used and all others are ignored. The *netid* and *luname* values are ignored once the NVSS command authorization table is loaded. The following example shows how generic characters cause the second command identifier to be ignored. If the following statements are included in the NVSS command authorization table for domain MSO01, only the first is used:

```
PROTECT *.*.AUTOTBL.MEMBER.DSITBL01
PROTECT *.MSO01.AUTOTBL.MEMBER.DSITBL01
```

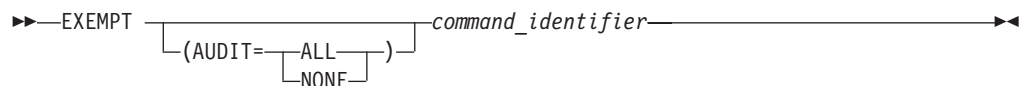
EXEMPT Statement: The EXEMPT statement identifies a command and optionally a keyword and value to be exempted from command authorization.

It enables all users to issue a command, keyword, or value, which is similar to defining a resource in the NETCMDS class with a universal access of read (UACC(READ)).

Using specific EXEMPT statements can reduce the amount of processing required for command authorization checking, and can improve performance.

The syntax for the EXEMPT statement is:

EXEMPT



Where:

command_identifier

Is the identifier specifying the *netid*, *luname*, command, keyword, and value to be exempted. See "Command identifiers" on page 196 for information on specifying command identifiers.

Command authorization

AUDIT

Specifies whether an audit record should be created when a command authority check yields a match on the command identifier. The audit records can be SMF type 38 records, or the DSIXITXL exit can write the records to an external log. The AUDIT keyword is optional. If not specified, auditing is determined by the value of CATAUDIT on the DEFAULTS command. When specified, the value overrides the value specified for CATAUDIT on the DEFAULTS command. The values allowed for AUDIT are:

ALL

Specifies that an audit record is to be created when a match occurs on the command identifier.

NONE

Specifies that no audit record is to be created when a match occurs on the command identifier.

Example:

To define a command identifier to exempt the LIST command in any domain, you must use the following statements:

```
EXEMPT *.*.LIST
EXEMPT *.*.LIST.*
```

The first statement applies only to the LIST command. The trailing asterisk in the second statement causes this command identifier to apply to all keywords and values of the LIST command that are not more explicitly specified.

GROUP statement: The GROUP statement defines a list of operators to be associated with a specific group name for command security purposes. The group name is unrelated to other groups of operators, such as the groups used to route messages using the NVSS ASSIGN command.

The syntax for the GROUP statement is:

GROUP

►► GROUP *group_name* — *userid* —►



Where:

group_name

Is the 1–8 character name of the group you are defining. The *group_name* cannot contain an ampersand (&), asterisk (*), or percent sign (%). The group name cannot be the same as any of your user IDs that are defined in the NVSS command authorization table.

userid

Is the 1–8 character identifier of a user to be included in the group. The *userid* cannot contain an ampersand (&), asterisk (*), or percent sign (%). This must be an individual user ID and not the name of a group.

Example:

To define a group named NIGHTOPS containing operators FELIX, MORRIS, and TOM, use the following:

```
GROUP NIGHTOPS FELIX,MORRIS,TOM
```

To define a large number of operators to a group, you can either repeat the same group name on multiple group statements or create a multiple-line group statement using the NVSS <BEGIN> and <END> statements.

PERMIT statement: The PERMIT statement authorizes a user ID or group to issue a command and optionally a keyword and value. The command identifier must have been previously protected with a PROTECT statement. You can include more than one PERMIT statement for the same command identifier.

The syntax for the PERMIT statement is:

PERMIT

```
➤—PERMIT authorized_name command_identifier—➤
```

Where:

authorized_name

Is the 1–8 character name of a user ID or a group that is authorized to issue the command, keyword, and value identified by the *command_identifier*. The *authorized_name* cannot contain an ampersand (&), asterisk (*), or percent sign (%). No checking is done to verify that a user ID is a valid NVSS operator ID.

Note: User IDs used in your table statements are independent of DSIOFP operator definitions and SAF product definitions. Even if an operator has been deleted from DSIOFP or the SAF product, the operator will continue to have the same command authority with respect to the active NVSS command authorization table as long as the operator remains logged on.

command_identifier

Is the identifier specifying the *netid*, *luname*, command, keyword, and value to be authorized. See “Command identifiers” on page 196 for information on specifying command identifiers.

Examples of generic characters in PERMIT and PROTECT statements: The following examples assume you are using the NVSS command authorization table statements to define command authorization and that your NVSS domain name (*luname*) is MS001. To authorize only NETOP1 to issue the GETCONID command with the AUTH keyword and a value of MASTER, include the following statements:

```
PROTECT *.MS001.GETCONID.AUTH.MASTER
PERMIT NETOP1 *.MS001.GETCONID.AUTH.MASTER
```

- To protect all other keywords on the GETCONID command, include the following statement:

```
PROTECT *.MS001.GETCONID.*
```
- It is not required to use a generic character in the value position, but since all of the GETCONID keywords have corresponding values, a command identifier of *.MS001.GETCONID.*.* would be functionally equivalent.
- Some commands have keywords that are issued without a corresponding value. For example, the SAVE and PPT keywords of the EVERY command do not have

Command authorization

a value. To authorize NETOP1 in domain MSO01 to issue the EVERY command with both the SAVE and PPT keywords, include the following statements:

```
* PROTECT KEYWORDS ON "EVERY" COMMAND
PROTECT *.MSO01.EVERY.SAVE
PROTECT *.MSO01.EVERY.PPT
PERMIT NETOP1 *.MSO01.EVERY.SAVE
PERMIT NETOP1 *.MSO01.EVERY.PPT
```

- To protect both the SAVE and PPT keywords and all other keywords on the EVERY command, include the following statement:
PROTECT *.MSO01.EVERY.*
- Notice that there is no generic character used for value. The command identifier *.MSO01.EVERY.* would not protect the SAVE and PPT keywords, but would only protect keywords that are specified with a corresponding value. The command identifier *.MSO01.EVERY.* protects keywords that have corresponding values as well as keywords that do not have corresponding values.
- Using an asterisk (*) as a trailing generic character at the end of a command identifier allows you match on subsequent values in that field and subsequent fields. Using a trailing asterisk in the *command* field will protect the command, and all its keywords and values. For example, if you use this statement:

```
PROTECT *.*.STOP*
```

It will protect the NVSS STOP command, and all its keywords and values. Note that this is equivalent to coding all three of the following statements:

```
PROTECT *.*.STOP
PROTECT *.*.STOP.*
PROTECT *.*.STOP*.*
```

- Using a trailing asterisk in the *keyword* field will protect the keyword for that command, with all the values on that keyword. For example, to protect all the values on all the REXX keywords for the NVSS DEFAULTS command, use this statement:

```
PROTECT *.*.DEFAULTS.REXX*
```

SETVAR statement: The SETVAR statement defines a table variable to represent multiple values which can be used in command identifiers. Table variables must represent an entire field value and must be defined before being used.

The syntax for the SETVAR statement is:

SETVAR

►► SETVAR *variable_name* *value* ◄◄



Where:

variable_name

Is the 1–32 character name of the variable you are defining. The variable name cannot contain an ampersand (&), dash (-), period (.), asterisk (*), or percent sign (%).

value

Is the 1–242 character value to be included in the command identifier. The value cannot contain an ampersand (&), dash (-), or period (.).

Examples:

Command authorization

To define a variable EURODOM to represent domains MS001, CNM02, and CNM99, use the following:

```
SETVAR EURODOM MS001,CNM02,CNM99
```

To subsequently use the variable &EURODOM in a PROTECT statement, include the following:

```
PROTECT *.&EURODOM.STOP
```

When processed, this generates the equivalent of the following table statements:

```
PROTECT *.MS001.STOP
PROTECT *.CNM02.STOP
PROTECT *.CNM99.STOP
```

Note that the table variable EURODOM represented the entire field value. A specification such as the following is *not* valid:

```
SETVAR EURODOM 01,02,99
PROTECT *.CNM&EURODOM.STOP
```

To define a variable XDOM to represent commands ROUTE and RMTCMD, use the following:

```
SETVAR XDOM ROUTE,DSIUSNDM
```

To subsequently use the variable &XDOM in a PROTECT statement, include the following:

```
PROTECT *.MS001.&XDOM
```

This generates the equivalent of the following table statements:

```
PROTECT *.MS001.ROUTE
PROTECT *.MS001.DSIUSNDM
```

Loading the NVSS command authorization table: The NVSS command authorization table can be loaded during NVSS initialization as specified by the OPTIONS statement in the DSIDMN member of DSIPARM. During initialization, if syntax errors are encountered, messages are issued but any valid statements in the table are still loaded. After NVSS initialization is complete, errors can be corrected and the table reloaded using the REFRESH command. If there are syntax errors in the table processed by the REFRESH command, the table is not loaded. There is a TEST keyword on the REFRESH command that you can use to check for syntax errors before attempting to load the table.

Using the NVSS command authorization table to protect VTAM command keywords and values: You can restrict any keywords and values of a VTAM command using the NVSS command authorization table. For values entered with the VTAM keyword ID, SLU, PLU, LU1, and LU2, if the VTAM resource is qualified with a network ID, access to the network ID and resource name are checked separately. So they should be defined in separate PROTECT statements. The VTAM resource name and the network ID can be up to 8 characters long. If IDTYPE=IPADDR is entered with the VTAM DISPLAY command, the value entered with the ID keyword is an IP address and can be longer than 8 characters.

For example, the following statements are defined in the NVSS command authorization table:

```
PROTECT *.*.DISPLAY.ID.NETA
PROTECT *.*.DISPLAY.ID.DSICRTR
PROTECT *.*.DISPLAY.ID.87/123/136/121
```

Command authorization

An NVSS operator would not be able to execute the following VTAM commands:

```
D NET,ID=NETA.NTVB5LU
D NET,ID=NETA.DSICRTR
D NET,ID=NETB.DSICRTR
D NET,ID=87.123.136.121,IDTYPE=IPADDR
```

The operator can, however, execute the following commands:

```
D NET,ID=NTVB5LUC
D NET,ID=NETB.NTVB5LUC
```

Command authorization table – usage notes: Some command identifiers are more specific than others. For example, the following table statements are ordered from most specific to least specific, as you can determine by comparing the character strings from left to right:

```
PROTECT *.MS001.STOP.FORCE.MS001PPT
PROTECT *.MS001.STOP.FORCE.*
PROTECT *.MS001.STOP.*
```

The most specific PROTECT statement in your NVSS command authorization table is the statement with the generic character latest in the sequence of fields, after the *netid* and *luname* fields. Only the most specific statement that matches the command being issued is used for command authorization.

The type of generic character is also used to determine which command identifier is most specific. Because the percent sign (%) generic character replaces just a single character, the percent sign is considered more specific than the asterisk (*) generic character. For example, ABC% is more specific than ABC* when evaluating the value ABCD.

For example, the value SYS1 matches both the SY%1 and the SYS* identifiers. In this case, SYS* is considered to be more specific because the generic character is in the fourth position, rather than SY%1 which has a generic character in the third position.

If both a PROTECT and an EXEMPT statement are coded for the same command identifier, message BNH184E will be issued indicating a syntax error in the NVSS command authorization table.

If this message is issued due to a REFRESH command, the NVSS command authorization table is not loaded. If the message is issued during initialization, the NVSS command authorization table is loaded, but only the first (PROTECT or EXEMPT) statement is used. Use message BNH184E to find the problem.

Command authorization table example: The following steps provide an example of defining operator authority using an NVSS command authorization table:

1. Define groups of operators.

```
GROUP GRP1 NETOP1,NETOP2,AUT01,AUT02
<BEGIN>
GROUP GRP2 OPER1,OPER2,OPER3,OPER4,OPER5,OPER6,NETOP1,NETOP2,
AUT01,AUT02
<END>
```

Note that these operators have been grouped into two classes of authorization.

2. Define the commands, keywords, and values to be protected.

- The following statements define the **OVERRIDE** command as unprotected except for the **REXXSTRF** keyword. This keyword can only be used by operators in group **GRP1**.

```
EXEMPT      NETA.MS001.OVERRIDE
PROTECT     NETA.MS001.OVERRIDE.REXXSTRF.*
PERMIT GRP1 NETA.MS001.OVERRIDE.REXXSTRF.*
```

- All of the following statements are comments. If you remove the asterisks from these statements, they protect the **GLOBALV** command and restrict its use to operators in groups **GRP1** and **GRP2**. The statements also protect the **SAVEC** and **RESTOREC** keywords, and restrict their use to operators in groups **GRP1** and **GRP2**. Finally, the statements protect the asterisk (*) and **PURGEC** keywords, and restrict their use to operators in group **GRP1**.

```
* PROTECT      NETA.MS001.GLOBALV
* PERMIT GRP1  NETA.MS001.GLOBALV
* PERMIT GRP2  NETA.MS001.GLOBALV
* PROTECT      NETA.MS001.GLOBALV.SAVEC
* PERMIT GRP1  NETA.MS001.GLOBALV.SAVEC
* PERMIT GRP2  NETA.MS001.GLOBALV.SAVEC
* PROTECT      NETA.MS001.GLOBALV.RESTOREC
* PERMIT GRP1  NETA.MS001.GLOBALV.RESTOREC
* PERMIT GRP2  NETA.MS001.GLOBALV.RESTOREC
* PROTECT      NETA.MS001.GLOBALV.ASTERISK
* PERMIT GRP1  NETA.MS001.GLOBALV.ASTERISK
* PROTECT      NETA.MS001.GLOBALV.PURGEC
* PERMIT GRP1  NETA.MS001.GLOBALV.PURGEC
```

Determining the user identity used for authority checking commands

Authority checking restricts the ability of an operator or a task to use commands, keywords, and values.

Table 4 on page 57 identifies the operator or task identifier that is used to authority check **msys** for Operations commands based on the command and environment. The identity is referred to as the **SOURCEID**.

Table 14. *SOURCEID* determination

Command and environment	SOURCEID determination
EXCMD command or a same-domain LABEL command prefix used to queue an imbedded command to another task.	The SOURCEID is the task that issued the EXCMD command, or the existing SOURCEID at the time the EXCMD command was issued.
TIMER commands that are scheduled to run under the PPT.	The SOURCEID is the task that issued the AT, EVERY, CHRON, or AFTER command, or the existing SOURCEID at the time the AT, EVERY, CHRON, or AFTER command was issued. Note: The SOURCEID is not destroyed by saving and restoring timer commands.
NVSS SUBMIT command for jobs submitted to the operating system from NVSS.	If OPERSEC=SAFCHK or OPERSEC=SAFDEF, the identity that is checked by the operating system is the issuer of the SUBMIT command, or the existing SOURCEID at the time the SUBMIT command was issued. For other values of OPERSEC, NVSS's authority is used for submitting the job.

Command authorization

Table 14. SOURCEID determination (continued)

Command and environment	SOURCEID determination
NVSS commands that were entered at an MVS operator console.	<p>When an MVS console has been associated with an autotask using the AUTOTASK command with the CONSOLE= parameter, NVSS commands can be entered from that MVS console. This is done by prefixing the NVSS command with the NVSS designator character, which by default is %. If the MVS operator has logged on to the MVS console with a user ID, the SOURCEID is the user ID of the MVS operator.</p> <p>If an operator has not logged on at the EMCS console, the SOURCEID of that task defaults to *BYPASS*. Commands issued from a source ID of *BYPASS* are not checked for command authorization by:</p> <ul style="list-style-type: none"> • The NVSS command authorization table • The SAF product OPERCMDS class • The SAF product NETCMDS class <p>Note: If a command is entered from the MVS master console, it will be routed to one of the following:</p> <ul style="list-style-type: none"> • The autotask with the specific console name • The autotask with console name <code>"*MASTER*"</code> • The autotask with console name <code>"*ANY*"</code>
NVSS commands that are entered using the MVS MODIFY command.	<p>When an MVS console has been associated with an autotask using the AUTOTASK command with the CONSOLE= parameter, NVSS commands can be entered from that MVS console by issuing an MVS MODIFY or STOP command against the NVSS task. The NVSS command is entered as text following the MODIFY command. The first parameter on the MODIFY command is the application ID that is being modified. If the MVS operator has logged on to the MVS console with a user ID, the SOURCEID is the user ID of the MVS operator.</p> <p>If an operator has not logged on at the EMCS console, the SOURCEID of that task defaults to *BYPASS*. Commands issued from a source ID of *BYPASS* are not checked for command authorization by:</p> <ul style="list-style-type: none"> • The NVSS command authorization table • The SAF product OPERCMDS class • The SAF product NETCMDS class <p>Note: If a command is entered from the MVS master console, it will be routed to:</p> <ul style="list-style-type: none"> • The autotask with the specific console name • The autotask with console name <code>"*MASTER*"</code> • The autotask with console name <code>"*ANY*"</code>
NVSS commands that were entered by TSO users.	<p>When a TSO user ID has been associated with an autotask using the AUTOTASK command with the CONSOLE= parameter, NVSS commands can be entered from that TSO user ID when the user is acting as an MVS operator by using an EMCS console session, or when using SDSF. The SOURCEID is the TSO user's user ID.</p>
Commands issued from JCL.	<p>When a job that issues a NVSS command is submitted by a TSO user ID, the SOURCEID is the TSO user ID. If the ID of the submitter is unknown, a default user ID is inserted. The value of the default user ID is defined by the system installation.</p>
MVS ROUTE command issued from NVSS.	<p>If the MVS command ROUTE is issued from a NVSS task, the originating source ID is always passed to the SAF product for authorization checks in the OPERCMDS class. This occurs for all settings of AUTHCHK and CMDAUTH.</p>

Table 14. SOURCEID determination (continued)

Command and environment	SOURCEID determination
Commands that are routed to an operator from the automation table.	The SOURCEID is the operator ID to which the command is routed. Note: Commands from the automation table are subject to authority checking unless SEC=BY was specified on the CMDMDL statement or SEC=DE was specified (or SEC was not specified) and AUTOSEC=BYPASS is in effect. For more information, refer to the DEFAULTS command in the NVSS online help.
CNMSMSG service (PL/I and C).	If CNMSMSG is called to queue a command from one task to another, the SOURCEID is the task name (TVBOPID) of the CNMSMSG issuer, or the existing SOURCEID at the time the CNMSMSG service was called.

Understanding security for specific commands

This section provides additional information about protecting the following commands:

- CHRON
- EXCMD
- MVS
- SUBMIT

Defining security for the CHRON command: The CHRON command has syntax that is more complex than most commands. CHRON uses multiple levels of keywords, items in lists, and quoted strings.

Command security for the CHRON command is checked so that operands within parentheses can be uniquely defined in the command authorization table (CMDAUTH=TABLE).

The following rules describe CHRON commands and which command identifiers are checked:

RULE 1: Each keyword that does not take a value (NOSAVE, SAVE, LOCAL, GMT, REFRESH, TEST, and DEBUG) is checked in the form:

Command example:

```
netid.luname.CHRON.keyword
```

RULE 2: Each keyword with a value is checked in the form:

Command example:

```
netid.luname.CHRON.keyword.value
```

With the CHRON command, the value may be a list or quoted string.

Command example:

```
CHRON AT=(),RECOVERY=IGNORE,NOSAVE,LOCAL,ROUTE=OPER1,ID=TEST1,COMMAND='MSG ALL HELLO'
```

The following command identifiers are checked:

```
netid.luname.CHRON
netid.luname.CHRON.AT.()
netid.luname.CHRON.RECOVERY.IGNORE
netid.luname.CHRON.NOSAVE
netid.luname.CHRON.LOCAL
```

Command authorization

```
netid.luname.CHRON.ROUTE.OPER1
netid.luname.CHRON.ID.TEST1
netid.luname.CHRON.COMMAND.'MSG_ALL_HELLO'
```

Rule 3A: Keywords appearing within parenthesized lists of other keywords are checked using the hierarchy of keywords with a "(" between so that the keyword hierarchy can be uniquely identified. The compound keyword that is generated is tested with the value of the innermost keyword. This checking is done at each level of the nesting of the lists. When a keyword is within a list that is the value of another keyword, the notation uses both keywords with a "(" between them.

Rule 3B: From the outermost to innermost, if a "keyword=(list)" appears, if any values appear in the list without keywords, the "keyword=value" check is done for that value. The keyword that is checked is the keyword hierarchy defined by Rule 3A.

Command example:

```
CHRON EVERY=(INTERVAL=(000-01.00.00 FOR=08.00.00))
```

The following command identifiers are checked:

```
netid.luname.CHRON
netid.luname.CHRON.EVERY.(INTERVAL=(000_01/00/00_FOR=08/00/00))
netid.luname.CHRON.EVERY(INTERVAL.(000_01/00/00_FOR=08/00/00))
netid.luname.CHRON.EVERY(INTERVAL.000_01/00/00
netid.luname.CHRON.EVERY(INTERVAL(FOR.08/00/00
```

Substitution of certain special characters is performed as described in "Protecting commands containing special characters" on page 56. For example, a dash becomes an underscore in the command identifier.

Rule 4: Quoted string values are checked as a single value, including the apostrophes and all text within the apostrophes.

Command example:

```
netid.luname.CHRON.REM.'ISN'T THIS A REMARK STRING?'
```

The following command identifier is checked:

```
CHRON REM='ISN'T THIS A REMARK STRING?'
```

Rule 5: For the DAYSWEK keyword, days of the week can be followed by a sublist identifying particular weeks of the month. The day name and each item in the sublist are treated as a unit.

Command example:

```
CHRON EVERY=(DAYSWEK=(NOT MON(1ST 2nd)))
```

The following command identifiers are checked:

```
netid.luname.CHRON
netid.luname.CHRON.EVERY.(DAYSWEK=(NOT_MON(1ST_2ND)))
netid.luname.CHRON.EVERY(DAYSWEK.(NOT_MON(1ST_2ND))
netid.luname.CHRON.EVERY(DAYSWEK.NOT
netid.luname.EVERY(DAYSWEK.MON(1ST)
netid.luname.EVERY(DAYSWEK.MON(2ND)
```

This lets you check the sublist values without concern for the order of the items within the sublist. Notice that the value "MON(1st 2nd)" is not checked since the values MON(1st) and MON(2nd) are checked.

Command authorization

The following table illustrates a detailed list of possible command identifiers that may be defined for the CHRON command. The rule that causes the command identifier to be checked is shown in the second column.

Table 15. NVSS command identifiers for the CHRON command

Commands and keywords identifier	RULE	Command authorization table identifier
CHRON	Command name	netid.luname.CHRON
AT=	2 2 3B 3B 2	netid.luname.CHRON.AT() netid.luname.CHRON.AT.(timespec datespec) ¹ netid.luname.CHRON.AT.timespec netid.luname.CHRON.AT.datespec ¹ netid.luname.CHRON.AT.yyy_mm_dd_hh/mm/ss/micros ¹
AFTER=	2 2	netid.luname.CHRON.AFTER.timespec ¹ netid.luname.CHRON.AFTER.ddd_hh/mm/ss/micros ¹
EVERY=	2 2 2	netid.luname.CHRON.EVERY.NONE netid.luname.CHRON.EVERY.() netid.luname.CHRON.EVERY.(everyoptions) ¹
EVERY=(INTERVAL=	3A 3B 3B 3A	netid.luname.CHRON.EVERY(INTERVAL.() netid.luname.CHRON.EVERY(INTERVAL.(intervaloptions) ¹ netid.luname.CHRON.EVERY(INTERVAL.timespec ¹ netid.luname.CHRON.EVERY(INTERVAL.ddd_hh/mm/ss/micros ¹
EVERY=(INTERVAL=(FOR=	3A 3A	netid.luname.CHRON.EVERY(INTERVAL(FOR.timespec netid.luname.CHRON.EVERY(INTERVAL(FOR.hh/mm/ss/micros ¹
EVERY=(INTERVAL=(MXREPEAT=	3A 3A	netid.luname.CHRON.EVERY(INTERVAL(MXREPEAT.NOLIMIT netid.luname.CHRON.EVERY(INTERVAL(MXREPEAT.repeat_count
EVERY=(INTERVAL=(OFF=	3A 3A	netid.luname.CHRON.EVERY(INTERVAL(OFF.timespec netid.luname.CHRON.EVERY(INTERVAL(OFF.hh/mm/ss/micros ¹
EVERY=(REMOVE=	3A 3A, 3B 3B 3B 3A	netid.luname.CHRON.EVERY(REMOVE.MANUALLY netid.luname.CHRON.EVERY(REMOVE.(removeoptions) ¹ netid.luname.CHRON.EVERY(REMOVE.datespec ¹ netid.luname.CHRON.EVERY(REMOVE.timespec ¹ netid.luname.CHRON.EVERY(REMOVE.yyyy_mm_dd_hh/mm/ss/micros ¹
EVERY= (REMAFTER=	3A 3A	netid.luname.CHRON.EVERY(REMAFTER.timespec ¹ netid.luname.CHRON.EVERY(REMAFTER.ddd_hh/mm/ss/micros ¹

Command authorization

Table 15. NVSS command identifiers for the CHRON command (continued)

Commands and keywords identifier	RULE	Command authorization table identifier
EVERY= (DAYSWEEK=	3A 3B 3B 3B 5	netid.luname.CHRON.EVERY(DAYSWEEK.ALL netid.luname.CHRON.EVERY(DAYSWEEK.(daysweeklist) ¹ netid.luname.CHRON.EVERY(DAYSWEEK.NOT netid.luname.CHRON.EVERY(DAYSWEEK.dayname netid.luname.CHRON.EVERY(DAYSWEEK. dayname (sublist_element) ¹
EVERY=(DAYSMON=	3A 3B 3B 3B	netid.luname.CHRON.EVERY(DAYSMON.ALL netid.luname.CHRON.EVERY(DAYSMON.(dayslist) ¹ netid.luname.CHRON.EVERY(DAYSMON.NOT netid.luname.CHRON.EVERY(DAYSMON.dayofmonth ¹
EVERY=(CALENDAR=	3A 3B 3B 3B	netid.luname.CHRON.EVERY(CALENDAR.ALL netid.luname.CHRON.EVERY(CALENDAR.(calendarlist) ¹ netid.luname.CHRON.EVERY(CALENDAR.NOT netid.luname.CHRON.EVERY(CALENDAR.keyname ¹
RECOVERY=	2 2 2	netid.luname.CHRON.RECOVERY.IGNORE netid.luname.CHRON.RECOVERY.AUTOLGN netid.luname.CHRON.RECOVERY.PURGE
SAVE	1	netid.luname.CHRON.SAVE
NOSAVE	1	netid.luname.CHRON.NOSAVE
LOCAL	1	netid.luname.CHRON.LOCAL
ID=	2	netid.luname.CHRON.ID.idname
NOTIFY=	2	netid.luname.CHRON.NOTIFY.(notifylists)
NOTIFY=(PURGE=	3B 3B	netid.luname.CHRON.NOTIFY(PURGE.(purgelist) netid.luname.CHRON.NOTIFY(PURGE.taskname
NOTIFY=(REMOVE=	3B 3B	netid.luname.CHRON.NOTIFY(REMOVE.(removelist) netid.luname.CHRON.NOTIFY(REMOVE.taskname
NOTIFY=(IGNORE=	3B 3B	netid.luname.CHRON.NOTIFY(IGNORE.(ignorelist) netid.luname.CHRON.NOTIFY(IGNORE.taskname
NOTIFY=(RUN=	3B 3B	netid.luname.CHRON.NOTIFY(RUN.(runlist) netid.luname.CHRON.NOTIFY(RUN.taskname
REFRESH	1	netid.luname.CHRON.REFRESH
TEST	1	netid.luname.CHRON.TEST
DEBUG	1	netid.luname.CHRON.DEBUG
COMMAND=	4	netid.luname.CHRON.COMMAND.'quoted string' ¹
REM=	4	netid.luname.REM.'quoted string' ²

Defining EXCMD command authorization: The NVSS EXCMD command is used to send commands to another task.

There are two operands that are used when issuing the EXCMD command. One is the *operator_id* where the command is being sent, and the other is the *command* being sent. These two operands are checked as a keyword-value pair.

Note: When protecting the target verb of EXCMD, specify the command verb, not any synonym. Unless otherwise documented, the verb is the label used on the CMDMDL statement. The verb for labeled commands beginning with a slash is EXCMD

For example, the command identifier to protect EXCMD OPER1 LOGOFF is:

```
PROTECT *.*.EXCMD.OPER1.LOGOFF
```

Defining additional MVS command authority: You can protect individual MVS system commands from unauthorized use with the OPERCMDS class of an SAF product, such as RACF. This is additional authorization checking done at the MVS level, after the command security checking done by the NVSS command authorization table.

To protect MVS commands:

1. Ensure your OPERSEC setting has a value of SAFCHECK or SAFDEF.
2. Define command profiles to restrict specific commands from operators. For example, to restrict all operators from being able to issue an MVS QUIESCE command, enter:

```
RDEFINE OPERCMDS MVS.QUIESCE UACC(NONE)
```

3. Ensure that the OPERCMDS class is active and enabled for processing. The following RACF commands can be used to do this:

```
SETROPTS CLASSACT(OPERCMDS)  
SETROPTS RACLIST(OPERCMDS)
```

4. When the OPERCMDS class is active, use the RACF REFRESH function when you change a definition:

```
SETROPTS RACLIST(OPERCMDS) REFRESH
```

Defining SUBMIT command authorization: You can protect jobs submitted from NVSS using the SUBMIT command. When the NVSS SUBMIT command is issued, the SUBMIT command can be protected using NVSS command authorization. By protecting at this level, you can stop the processing for unauthorized users before the job is ever submitted to the system.

Controlling access to data sets and members

To prevent unauthorized alteration of data, you can protect data sets with an SAF product, such as RACF. To prevent unauthorized viewing of passwords and other restricted information, protect them with NVSS commands such as READSEC and WRITESEC.

Data set security

You can restrict unauthorized alteration of data sets from the NVSS environment using the DATASET class of the security product. The following are some considerations when using the DATASET class of the security product:

2. This value may have a special character, such as "." or "-", for example in the programmer time notation. You substitute the character "/" for "." and "_" for "-" when making the security definition.

Controlling access to data sets and members

- NVSS requires CONTROL access to the DSILog data set to write to the netlog.
- NVSS requires READ access to the first data set identified by the DSILIST DD statement.
- NVSS requires READ access to non-DSIPARM datasets that are specified on the NVSS SUBMIT command.
- Each of the following NVSS commands require UPDATE access to the first data set identified by the DSILIST DD statement.
 - AUTOTBL (with the LISTING keyword)
 - AUTOCNT (with the FILE keyword)
 - QRYGLOBL (with the FILE keyword)
 - SECMIGR (with output to DSILIST)
- If you use SECMIGR to convert from the NVSS command authorization table to RACF (TBL2RACF), the operator running SECMIGR requires READ authority to the data set containing the NVSS command authorization table being converted.
- If you specify your own output data set, the operator running SECMIGR requires UPDATE authority to your output data set.

Note: NVSS trace records are not made for calls to the DATASET class, because the calls are made by MVS for the NVSS tasks.

Restricting access to data sets

To activate the data set protection described in the preceding section, do the following:

1. To enable task-level authorization checking, initialize NVSS product using OPTIONS values of OPERSEC=SAFCHECK or OPERSEC=SAFDEF. If you did not initialize the NVSS product using these values, you can also change the OPERSEC values using the NVSS REFRESH command.
2. If you are using an SAF product, add profiles for the data sets you want to protect. The RACF product requires that the highest-level qualifier of the data set name be either a task or group name.

For example, use the RACF ADDSD command to add data set profiles. From an authorized TSO user, enter the following command to protect the OPER1.STATS data set:

```
ADDSD 'OPER1.STATS'
```

3. If you are using an SAF product, authorize the operator tasks so they can access the data set. For example, use the RACF PERMIT command to authorize operator tasks to the data set. To authorize NETOP1 to have update access to OPER1.STATS, enter the following command from an authorized TSO user:

```
PERMIT 'OPER1.STATS' CLASS(DATASET) ID(NETOP1) ACCESS(UPDATE)
```

NVSS READSEC and WRITESEC commands

Use the NVSS READSEC and WRITESEC commands to restrict access to data sets and members by NVSS commands. When you specify security for the READSEC command, it affects all of the NVSS commands which can display sensitive information, such as:

- BROWSE with a member name
- NCCF LIST with the CLIST or PROFILE keywords
- PIPE stages
 - < (From disk)
 - QSAM
- VSAM command DSIVSMX

Controlling access to data sets and members

Using READSEC and WRITESEC is the only way to prevent operators from viewing data sets and members using these NVSS commands. In NVSS, security is defined so that operators have access to DSIOPEN and msys for Operations online help. DSIOPEN is a DD name designed to hold information which should not be secured, such as NEWS data and PF key definitions. Anything other than DSIOPEN and the online help may be considered sensitive information.

Because attempts to define security for these NVSS commands is considered a severe error, message BNH115A is generated every time an operator logs on. The error text for this message is "SPECIAL SECURITY IN EFFECT FOR BROWSE AND READSEC", which indicates NVSS has defined default protection for sensitive data sets and members, and the NVSS commands which display data sets or members will fail. You must delete any security definitions for the commands and reinitialize NVSS to clear the error condition.

If you use command authorization without specifying values for READSEC and WRITESEC, operators will have access to all data sets and members.

Do not protect DD name CNMPNL1, operators need to access online help that is contained there.

For more information about how to use the READSEC and WRITESEC commands, refer to the online help.

Controlling access to data sets and members

Appendix B. Return codes

This Appendix provides information about macro return codes and about return codes from VIEW and BROWSE (see “VIEW and BROWSE return codes” on page 224).

Macro return codes

This section provides return codes for the following macros:

- DSICES
- DSIDKS
- DSIMQS
- DSIPRS
- DSIPSS
- DSIPUSH
- DSIZCSMS
- DSIZVSMS

Macro DSICES return codes in Register 15

The following return codes for macro DSICES are found in register 15:

- | | |
|----|--|
| 0 | The function is successful. One of the following describes what occurred: <ul style="list-style-type: none">• A regular command is found in the system command table and the address of the SCT entry is returned.• The verb is not found in the SCT (if you specify CLISTCK, a command list is found with the specified name), and the dummy SCT entry for a command list is returned. |
| 4 | The command that is found can be processed as a regular or immediate command; the address is returned. |
| 8 | An immediate command is found in the system command table; the address is returned. |
| 12 | The module is not found, or there is an incorrect verb length; no address is returned. |
| 16 | The operator is not authorized to issue the command. This is caused by the security definitions that are in place for this command. No address is returned in SCTADDR.

This return code is not applicable if MODNAME was specified. |
| 20 | Either the command found is incompatible with the task type that called the routine, and the address is returned; or you specified CLISTCK=YES and the request is issued in an asynchronous exit, and the address is not returned. |
| 24 | You specified CLISTCK=YES but the command or command list is not found in DSISCT or DSICLD. |
| 28 | You specified CLISTCK=YES but storage requested for CLISTCK processing is not obtained. |
| 32 | NVSS internal error. |

Macros

- 36 An unexpected return code was received from the security authorization facility (SAF). Message BNH238E is issued with the SAF return code inserted. This return code is not applicable if MODNAME was specified as this specification causes no authorization checking to be performed.
- 40 Authorization to the command is not granted because the security environment for the operator cannot be established. Message BNH239E is issued when this condition is first encountered to provide the security product return code information. Message BNH273I is issued when the condition has been corrected. This return code is not applicable if MODNAME was specified as this specification causes no authorization checking to be performed.
- 44 Authorization to the command is not granted because an unexpected return code was received from the command authorization table. Message BNH199E is issued indicating the command identifier and the operator ID being checked. This return code is not applicable if MODNAME was specified as this specification causes no authorization checking to be performed.
- 48 Authorization to the command is not granted because the NVSS internal security information containing the source ID of the command could not be found. Message BNH277E is issued identifying the command being checked. This return code is not applicable if MODNAME was specified as this specification causes no authorization checking to be performed.
- 52 Authorization to the command is not granted because the source ID is blank in the NVSS internal security information. Message BNH277E is issued identifying the command being checked. This return code is not applicable if MODNAME was specified because this specification causes no authorization checking to be performed.

Macro DSIDKS return codes in Register 15

The return codes and code meanings in register 15 are dependent on the "TYPE=" specification.

The following return codes are for TYPE=CONN:

- 0 The function is successful. Data control blocks and I/O buffer are obtained and initialized.
- 4 An incorrect data set name.
- 12 No storage was available for I/O buffer.

The following return codes are for TYPE=FIND:

- 0 The function is successful. The member or file is found and the first record is read.
- 4 The member or file is not found in the source statement library or in the specified library, or an empty member or file is found.
- 8 The member or file is found but an I/O error occurred on the first read.
- 12 The specified definition name or data set has not been opened.
- 20 The specified control block identifier is not valid; the member or file is not found.
- 28 There is a syntax error in the %INCLUDE card.

- 36 There is an incorrect member name on the %INCLUDE card.
- 40 There is an incorrect embed member, which can cause a deadlock condition. (This occurs when a member embeds itself.)
- 44 An unrecoverable system error occurred. An internal NVSS service failed, because of a storage failure.
- 46 An I/O error is encountered while trying to include a member specified in a %INCLUDE statement.

100 + xx

An error occurred during CLOSE processing. msys for Operations attempts to recover the data set after a failure during a previous FIND or READ. Refer to the description of the xx return code under the CLOSE macro in your operating system macro reference.

200 + xx

An error occurred during OPEN processing. msys for Operations attempts to recover the data set after a failure during a previous FIND or READ. Refer to the description of the xx return code under the OPEN macro in your operating system macro reference.

The following return codes are for TYPE=DISC:

- 0 The disconnect is successful; data and I/O buffers are freed successfully.
- 20 The specified control block identifier is not valid and no storage is freed.
- 46 An I/O error is encountered while trying to INCLUDE a member specified in a %INCLUDE statement.

The following return codes are for TYPE=READ:

- 0 The function is successful; the record is read.
- 4 The end of data is reached.
- 8 An I/O error occurred during reading.
- 12 Reading of this record is prohibited; an I/O error may have occurred, the end of data may have been reached, or the caller did not issue TYPE=FIND first.
- 20 The specified control block identifier is not valid; the record is not read.
- 28 There is a syntax error in the %INCLUDE card.
- 36 A member name on the %INCLUDE card is not valid.
- 40 An embed member is not valid and can cause a deadlock condition. (This occurs when a member embeds itself.)
- 44 An unrecoverable system error occurred. An internal service failed, because of a storage failure.
- 46 An I/O error is encountered while trying to include a member specified in a %INCLUDE statement.

100 + xx

An error occurred during CLOSE processing. msys for Operations attempts to recover the data set after a failure during a previous FIND or READ. Refer to the description of the xx return code under the CLOSE macro in your operating system macro reference.

200 + *xx*

An error occurred during OPEN processing. msys for Operations attempts to recover the data set after a failure during a previous FIND or READ. Refer to the description of the *xx* return code under the OPEN macro in your operating system macro reference.

Macro DSIMQS return codes in Register 15

The following return codes for macro DSIMQS are found in register 15:

- | | |
|----|---|
| 0 | The function was successful; the message is queued. |
| 4 | The buffer length was either: <ul style="list-style-type: none">• Not greater than 0• Less than the combined length of HDRBLEN plus HDRTDISP• Greater than 32000 |
| 8 | The operator ID designated as the receiver of authorized messages was not found. |
| 12 | A buffer could not be obtained or dynamic resource control failed. |
| 16 | msys for Operations is terminating; the external request cannot be completed. |
| 20 | The SWB address is not valid. |
| 22 | The list specified with the LIST option contained no operator IDs. It contained only unassigned group IDs. |
| 23 | Messages were routed to the first 255 operators or groups, or both. |
| 24 | The value specified for priority was not valid. |
| 26 | The internal function request for the command to be run contains the IFRAUTBC or IFRAUTBN fields. The task that receives this command has no MQS receipt support and cannot process these fields. |
| 28 | A message stack enquiry failed. |
| 32 | NVSS internal error. |

Note: When a command procedure written in REXX or NVSS command list language is executing, NVSS services all message queues, except the low-priority queue, at three points:

- Initially, before the first instruction
- After the execution of any NVSS command
- Throughout the period of any wait state (for &WAIT, &PAUSE, or WAIT)

Because of this, two command lists queued at the same time to the high- or normal-priority queues appear to run in reverse order. The first one is initiated, then before it executes its first instruction, it is preempted, and the second command executes. To have command lists execute in the order queued, always queue them at low priority.

Macro DSIPRS return codes in Register 15

The following return codes for macro DSIPRS are found in register 15:

- | | |
|---|---|
| 0 | The function was successful. The required size of the table was returned in PDBSIZE, or the command was parsed and the parse table was built. |
| 4 | The input buffer was parsed, but there was no data in the input buffer (0 |

length data) or the data in the input buffer was all blanks. Only the buffer address and number of entries (0) could be returned in the parse table.

- 8 The parse table was too small for the input buffer; a partial parse table was built, and the number of entries was set to the number that the parse table could hold. The size of the parse table should be increased.
- 12 Unbalanced quotes. Returned only if SUB=YES is specified.
- 16 The number of characters between two consecutive delimiters in the input buffer was greater than 255.
- 20 An unpaired double-byte character set (DBCS) delimiter of DBCS data bytes was found in the input buffer. For example, one of the following may have occurred:
 - The end of the input buffer was found before the DBCS data-ending delimiter shift-in (X'0F').
 - A second DBCS data-beginning delimiter shift-out (X'0E') was found before the DBCS data-ending delimiter shift-in (X'0F').
 - An odd number of DBCS data bytes were found between DBCS data delimiters.
- 100 No PDB or an incorrect PDB was passed; or no PDBSIZE or an incorrect PDBSIZE was passed.

Note: You must specify the operands DELIM, FIRST, and SUB, identically, in the pair of DSIPRS parse commands issued. Otherwise, the second parse can fail or the storage can be overlaid.

Macro DSIPSS return codes in Register 15

The following return codes for macro DSIPSS are found in register 15:

- 0 The function was successful; the message is written. For TYPE=PSSWAIT, an ECB has been posted. Check the ECB list to determine which event has completed. For TYPE=ASYPANEL, the send or receive request has passed NVSS syntax and buffer checking and has been sent to VTAM; it does not indicate the success or failure of VTAM completion of the receive. You must check the ECB post code to determine the success or failure of the ASYPANEL request. The post code is put into the ECB specified in the panel parameter list.
- 4 For TYPE=XSEND, no request parameter list (RPL) was found and no data was sent.
- 8 Parameter error. There is an error in the formatting of the message buffer header. For TYPE=XSEND, the session is not active and no data is sent. For TYPE=ASYPANEL, the parameter list is inconsistent. If you specify the output buffer, you must also specify the length. If you specify the input ECB, you must specify the input area address, input area length, and the data length address of the returned length.
- 12 There is not enough storage available to complete the request. No output is sent, and the input command processor cannot be scheduled.
- 16 DSIPSS TYPE=OUTPUT was issued for an immediate command or in an IRB exit routine. Use DSIPSS TYPE=IMMED or DSIMQS instead.
- 20 No terminal session exists. For TYPE=ASYPANEL, the panel request came from a task other than an OST. No input is received. For TYPE=CANCEL, the panel request came from a task other than an OST.

Macros

- 36 For TYPE=ASYPANEL, a temporary error occurred. The contents of the panel have been modified. Reformat the panel using an Erase/Write or Erase/Write Alternate 3270 command. Then retry the request.
- 40 A permanent I/O error occurred. Do not retry the request. No output is sent, and no input processor is scheduled. For TYPE=ASYPANEL, no input is received. For TYPE=CANCEL, NVSS is unable to restart normal terminal activity.
- 48 For TYPE=ASYPANEL, no I/O is scheduled because the command processor issued a second DSIPSS TYPE=ASYPANEL requesting input before the previous request had completed.
- 56 For TYPE=PSSWAIT or TYPE=TESTWAIT, at least one NVSS ECB was posted.
- 68 For TYPE=OUTPUT or TYPE=IMMED, a message being processed for the RMTCMD command failed to be transmitted. This error can occur when the DSIUDST task is inactive.

The following ECB post codes for PSS TYPE=ASYPANEL are found in the ECB if you specified one:

- 0 The function was successful; the requested data is available.
- 12 There is not have enough storage available to complete the request. The output data was sent, but the input data is not available.
- 36 A temporary error occurred during a full-screen read. Retry the request. The output data was sent, but the input data is not available.
- 40 A permanent error occurred during a full-screen read. Do not retry the request. The output data was sent, but the input data is not available.
- 52 The requested input was canceled by DSIPSS TYPE=CANCEL. Do not retry the request immediately. The output data was sent, but the input data is not available.

Macro DSIPUSH return codes in Register 15

The following return codes for macro DSIPUSH are found in register 15:

- 0 The function was successful; the long-running command request is queued.
- 4 Storage is not available for the request.
- 8 The ABEND reinstate or LOGOFF routine is required but was not specified.
- 12 The request was issued from an incorrect task:
- RESUME request issued under DST
 - ABEND request issued under DST
 - DSIPUSH issued and task is not an OST, NNT, DST, or PPT
- 16 The request was issued while in an immediate command or while the NVSS program is in an exit, or in the middle of a LOGOFF routine or ABEND reinstate routine.
- 20 The RESUME routine is a command list, or the CMDMDL statement did not pass validity checking, or the operator's scope class does not permit access to the RESUME routine.

Verify that the first CMDMDL statement for this command in DSICMD is not type immediate.

- 24 The ABEND reinstate routine is a command list, or the CMDMDL statement did not pass validity checking, or the operator's scope class does not permit access to the RESUME routine.

Verify that the first CMDMDL statement for this command in DSICMD is not type immediate.
- 28 The LOGOFF routine is a command list, or the CMDMDL statement did not pass validity checking.

Verify that the first CMDMDL statement for this command in DSICMD is not type immediate.
- 32 The macro invocation is not valid. Fix assembly errors before trying to run the program.

Macro DSIZCSMS return codes in Register 15

The following major return codes for macro DSIZCSMS are found in register 15:

- 0 The function was successful; data was sent to VTAM.
- 4 The requested function could not be performed.
- 8 The input buffer was too small to build a forward RU.
- 12 An error was found in a parameter specification.
- 16 The program did not execute under a data services task.
- 20 The RULENG exceeded the maximum RU length required.

Macro DSIZCSMS minor return codes in Register 0

The following minor return codes for macro DSIZCSMS are found in register zero (0):

- 0 The function was successful.
- 4 The SWB was not valid.
- 8 The DSRB was not valid.
- 12 The DSRB that was passed was in use.
- 16 An unsolicited DSRB was passed.
- 20 An operator ID specified in the DSRB was not valid.
- 24 Reserved.
- 28 There was insufficient storage to process the request.
- 32 The CNMI is inactive.
- 36 The request was rejected by the access method.
- 48 The specified SECONDS value was not valid.

Macro DSIZVSMS return codes in Register 15

The following major return codes for macro DSIZVSMS are found in register 15:

- 0 Successful completion of VSAM function.
- 4 Manipulative macro error occurred during processing.
- 8 An error occurred in the EXECUTE form of a manipulative macro. An operand was not in the list.

Macros

- 12 Unsuccessful completion.
- 16 DSIZVSMS was issued while not executing under a DST.

Macro DSIZVSMS minor return codes in Register 0

The following minor return codes for macro DSIZVSMS are found in register zero (0):

- 0 Successful completion.
- 4 The specified DSRB was not valid or in use.
- 8 An ACB was unavailable or was not open. This may be due to a SWITCH *taskname*,T command having been issued.
- 12 Resume verb processing error.
- 16 An installation exit rejected the request.
- 20 The VSAM I/O request was not valid or there was an I/O scheduling error.
- 24 Data truncation occurred during substitution of data in an installation exit; or control block storage could not be obtained.
- 28 An installation exit returned a return code that was not valid.

Note: For more information about specifying FUNC and OPTION, refer to the OS/VS VSAM library. For an explanation of RPL feedback codes, refer to the OS/VS VSAM library and the MVS/ESA library.

VIEW and BROWSE return codes

Table 16 lists and describes the return codes that can be received for the VIEW and BROWSE command. The table also provides a brief description of the action you need to take.

Table 16. Return codes from VIEW and BROWSE

Code	Meaning	Your action
4	<ul style="list-style-type: none">Specified panel not found in CNMPNL1, CNMMSGF, or CNMCMDF data sets (MVS).Possible input/output (I/O) error.	Put panel definition in correct data set or file.
8	Panel definition format not valid; no noncomment lines found.	Correct format of panel definition.
12	You are not authorized to browse the member.	Ask your system programmer to redefine your authorization.
16	VIEW command processor invoked with parameters that are not valid. <i>Name1</i> must be 1 to 8 characters and <i>name2</i> must be a valid panel ID. Valid parameters are INPUT, NOINPUT, MSG, NOMSG.	Correct command list to use valid option.
24	Full-screen command processor is available to OST only.	Do not invoke VIEW from a non-OST.
28	Change file to logical record length of 80 bytes.	
32	Unrecoverable error resulted from macro call. Error could be that CNMMSGF or CNMCMDF has not been installed for online message or command help. Also, refer to message DWO050I in the NVSS log.	Install CNMMSGF or CNMCMDF. Otherwise, call IBM for service.

Table 16. Return codes from VIEW and BROWSE (continued)

Code	Meaning	Your action
36	Unrecoverable internal programming error occurred. Also, refer to message DWO050I in the NVSS log.	Call IBM for service.
40	Browse panel CNMBROWS, which is used for browsing members, was not found.	Put CNMBROWS in correct data set or file.
81	Panel definition format not valid; no text indicator line found, or more than 49 option definitions found.	Correct format of panel definition.
83	Panel definition format not valid; comment lines in wrong place.	Correct format of panel definition.

Macros

Appendix C. Coexistence of msys for Operations and SA OS/390 V1R3

When you migrate your system to z/OS V1R2, and you are using SA OS/390 V1R3, you can run msys for Operations and SA OS/390 V1R3 on the same systems if the following requirements are met:

- msys for Operations and SA OS/390 V1R3 must belong to different XCF groups. Since the XCF group suffix for msys for Operations should not be changed, this implies that the suffix for SA OS/390 V1R3 must not be A0.
- To ensure that msys for Operations and SA OS/390 V1R3 do not generate the same console names (for the requirement of unique console names see “EMCS console names and the automation router task” on page 25) you must change the global variable AOFCNMASK in AOFEXDEF (see *System Automation for OS/390 Customization Version 1 Release 3*, Appendix F). For example, you could specify AOFCNMASK=290C0D0E0F101518. The default is 290C0D0E0F101718.
- Certain functions of SA OS/390 V1R3 that are also available in msys for Operations must be switched off in SA OS/390 V1R3. These are:
 - WTO buffer recovery
To switch off WTO buffer recovery perform the following steps:
 1. Select the **MVS Component** policy object.
 2. Select the WTOBUF AUTOMATION policy of **MVS Component**.
 3. Set the **Recovery** flag to NO.
 - The CFDRAIN command
This command is available in msys for Operations as the DRAIN subcommand of the INGCF command. To disable the CFDRAIN command in SA OS/390 V1R3 remove the CFDRAIN synonym from AOFCMD.

Appendix D. Response Messages, Error Strings, Condition Codes

Response Messages (AOFA0000 – AOFA0018)

The automation returns the following messages to indicate command invocation problems, parameter list problems, or parameter resolution problems.

AOFA0000

This response message is returned as an indicator for command invocation problems, parameter list problems, or parameter resolve problems. If the AOFA0000 response message is returned from the INGHWCMD command list, its data portion is an error string. If the AOFA0000 response message is returned from the INGHWCOM communication task command processor, its data portion contains an condition code 001 - 033.

Example 1:

```
INGHWCMD MYSYS GETISTAT
```

```
AOFA0000 Resolve error:  
"MYSYS"_is_not_a_predefined_system_or_CF-name
```

The INGHWCMD command failed returning an AOFA0000 error string.

Example 2:

```
INGHWCMD SC50 GETISTAT
```

```
AOFA0000 GETISTAT STATUS(REJECTED) CONDITION(00B00003) SENSE() CPCSNAME()  
TSTIME(020111073708)
```

The INGHWCMD command failed. Message AOFA0000 was returned by INGHWCOM command processor. The condition code 00B00003 indicates that an unknown communication interface name was passed from INGHWCMD to INGHWCOM.

AOFA0001

This response message is returned from a request of the following hardware command functions:
ACTIVATE, DEACTIVATE, LOAD, RESTART, SYSRESET, START, STOP, CBU, EXTERNAL

Example 1:

```
INGHWCMD KEY7 STOP
```

```
AOFA0001 STOP KEY STATUS(SUCCESS)  
CPCSNAME(DEIBMD1.X7E1FA0A)TSTIME(020111135810)
```

The hardware function STOP was successfully performed for system KEY7.

Example 2:

```
INGHWCMD KEY6 SYSRESET
```

```
AOFA0001 SYSRESET KEY6  
CONDITION(00B00056) SENSE() CPCSNAME(DEIBMD1.X7E1FA0A) TSTIME(020111142827)
```

The hardware function SYSRESET was rejected by INGHWCOM. The condition code 00B00056 indicates that system KEY6 is still operational and cannot be disrupted. In order to perform a disruptive hardware operation, the FORCE option must be specified.

AOFA0002

This message is the response to an INITCOM request. INITCOM establishes a session between INGHWCOM and the Processor Support Element of the addressed hardware.

Example 1:

```
INGHWCMD YORAMA INITCOM  
  
AOFA0002 INITCOM YORAMA STATUS(SUCCESS)  
CPCSNAME(DEIBMD1.X7E1FA0A) TSTIME(020111143851)
```

The session between the INGHWCOM and the processor Support Element of the CPC DEIBMD1.X7E1FA0A, configured with the hardware name "YORAMA" is established successfully.

AOFA0004

This message is the response to a TERMCOM request. TERMCOM ends a session between INGHWCOM and the Processor Support Element of the addressed hardware.

Example 1:

```
INGHWCMD P701 TERMCOM  
  
AOFA0004 TERMCOM P701 STATUS(SUCCESS)  
CPCSNAME(USIBMSC.SCZP701) TSTIME(020111090930)
```

The session between the INGHWCOM and the Processor Support Element of the CPC USIBMSC.SCZP107 configured with the hardware name "P701" is terminated successfully.

Example 2:

```
INGHWCMD P701 TERMCOM  
  
AOFA0004 TERMCOM P701 STATUS(REJECTED) CONDITION(00B00033) SENSE()  
CPCSNAME(USIBMSC.SCZP701) TSTIME(020111091447)
```

The session termination between the INGHWCOM and the Processor Support Element of the CPC USIBMSC.SCZP701 configured with the hardware name "P701" was rejected. Condition code 00B00033 indicates that no session existed to terminate.

AOFA0017

This message is the response to a GETISTAT or GETSSTAT request. GETISTAT queries the status of an image object and GETSSTAT queries the status of a CPC object. On successful completion, the status field of msg AOFA0017 may have one of the following values:

GETISTAT	GETSSTAT
-----	-----
OPERATING	OPERATING
NOT_OPERATING	NOT_OPERATING
NOT_ACTIVATED	NO_POWER
STATUS_CHECK	STATUS_CHECK
EXCEPTIONS	EXCEPTIONS
POWERSAVE	POWERSAVE
	SERVICE
	LINKNOTACTIVE
	SERVICE_REQ

Example 1:

```
INGHWCMD P801 GETSSTAT

AOFA0017 GETSSTAT P801 STATUS(NO_POWER)
CPCSNAME(USIBMSC.SCZP801)TSTIME(020111095319)
```

The CPC USIBMSC.SCZP801 is currently powered off.

Example 2:

```
INGHWCMD * GETISTAT

AOFA0017 GETISTAT SC50
STATUS(OPERATING)CPCINAME(A3)CPCSNAME(USIBMSC.SCZP801)
TSTIME(020111095940)
```

The own system (*), which runs on LPAR A3 of CPC USIBMSC.SCZP801 has a status of OPERATING and its system name is SC50.

Example 3:

```
INGHWCMD YORAMA.KEY7 GETISTAT

AOFA0017 GETISTAT YORAMA STATUS (OPERATING) CPCINAME(KEY7)
CSNAME(DEIBMD1.X7E1FA0A) TSTIME(020204130403)
```

On processor YORAMA, the logical partition KEY7 has a status of OPERATING.

AOFA0018

Example 1:

```
INGHWCMD * GETCLUSTER

AOFA0018 GETCLUSTER SC50 STATUS(SUCCESS)
CPCSNAME(USIBMSC.SCZP801)TSTIME(020112054842)
AOFA0018 GETCLUSTER PDATA(USIBMSC.SCZP801)
AOFA0018 GETCLUSTER PDATA(USIBMSC.SCZP701)
AOFA0018 GETCLUSTER PDATA(USIBMSC.SCZP702)
AOFA0018 GETCLUSTER PDATA(USIBMSC.SCZP601)
AOFA0018 GETCLUSTER REPORT COMPLETE
```

From the own system (*), the CPC addresses list in PDATA are in your scope of control. With an BCP internal interface connection, this list is determined internally from the local SE by contacting the HMC in your processor LAN which has the "Change Management" function enabled. The content of the Defined CPCs group of this HMC represents the CPCs you can contact through this BCP internal

interface session. Each scope list is terminated with a "report complete" message. The PDATA field of the AOFA0018 message contains CPC related information. The first PDATA entry is always the full qualified address of the CPC (cpcsname). Other PDATA information may be added in the future, separated by a comma.

Example 2:

```
INGHWCMD * GETCLUSTER

AOFA0018 GETCLUSTER SC50 STATUS(FAILED) CONDITION(0B100224)
        SENSE(00000000 0000 00000000)
        CPCSNAME(USIBMSC.SCZP801)TSTIME(020111085916)
```

This GETCLUSTER request failed with a condition code of 0B100224 representing a BCP internal interface transport timeout condition.

Error Strings

AOFA0000 Parm error:

```
Proc_or_Sys_name_and_HW_function_name_is_required
Timeout_range_ttt_already_defined
Timeout_tt_out_of_range_1-59.
Timeout_specification_range_tt_is_not_valid.
Timeout_specification_tt_is_not_valid.
Timeout_specification_ttt_ends_invalid.
hwcmd_with_FORCE_operand_is_not_valid.
hwcmd_is_not_a_supported_HW-function.
p_must_be_a_decimal_integer_value_or_ALL
p_EXTERNAL_CPU_definition_error
CN_Activation_profile_name_not_alphanumeric
CN_Profile_name_is_a_positional_parm
p_does_not_support_target_wildcard.
p_is_in_wrong_position.
lparm_load_parm_length_must_be_8.
devnum_device_address_not_hexadecimal.
devnum_mandatory_load_address_invalid.
lval_Load_value_definition_error.
lval_Load_value_is_a_positional_parm.
pn_load_profile_name_not_alphanumeric.
pn_Load_profile_definition_error.
pn_Profile_name_is_a_positional_parm.
spc_P_and_LV_specs_are_mutually_exclusive.
name_invalid_chars_in_proc_or_sys_name.
name_name_longer_than_8_characters.
parm_Parm_is_unknown_or_in_wrong_position.
```

AOFA0000 Resolve error:

```
sysname_is_not_related_to_a_hardware.
sysname_for_CFs_LOAD/SYSRESET_are_not_supported.
sysname_type_specification_missing_or_invalid.
pname_has_invalid_CPC_address_format.
pname_has_no_CPC_address_defined.
sysname_is_not_a_predefined_system_or_CF_name.
pname_processor_not_valid_for_CPC_command.
pname_is_HW.IMAGE_format_invalid_for_CPC_cmds
```

AOFA0000 Check Task:

```
hwtask_reached_QueueLimit_qlim
hwtask_task_msqueue_data_is_invalid
hwtask_task_is_not_available
```

AOFA0000 Hardware Interface:

internal_interface_is_disabled_or_not_active
internal_interface_status_cannot_be_determined

AOFA0000 Authorization error:

hwcmd_has_undefined_access_level
acclevel_to_resname_not_allowed_for_user
BadRC_during_access_chk_for_resname

Condition Codes

Hardware Communication Task Condition Codes "00B00xxx"

Reason Code	Error String	Error Description
001	ING_invalid_HLL_buffer	INGHWCOM was invoked, but the NetView HLL buffer found for C/C++ is not valid.
002	ING_origuser_invalid	The userid and output correlator passed to INGHWCOM is not valid.
003	ING_interface_invalid	The hardware interface name passed to INGHWCOM is not valid. Allowed interface names are BCP internal interface, SNMP, or SNA.
004	ING_interface_missing	No hardware interface name is passed to INGHWCOM.
005	ING_tgt_length-error	Parsing Error: The target object name (processor or image name) has an invalid length. Must be 1 - 8 characters.
006	ING_tgt_missing	Parsing Error: The target object name (processor or image name) is not specified.
007	ING_cpc_length_error	Parsing Error: The CPC address specification netid.nau has an invalid length. Length must not exceed 17 characters.
008	ING_cpc_missing	Parsing Error: The CPC address specification, which is a required parameter for the request is missing.
009	ING_imgname_length_err.	Parsing Error: The image name (Lpar name) parameter has an invalid length.
00A	ING_imgname_missing	Parsing Error: The image name (Lpar name) is a required parameter for the request, but is not specified.
00B	ING_force_invalid	Parsing Error: The FORCE option is specified in the request but is not supported for the HW function. The following HW functions allow the FORCE option: ACTIVATE, DEACTIVATE, SYSRESET, LOAD
00C	ING_force_missing	Parsing Error: The FORCE option is required for the request, but is not specified.
00D	ING_auth_missing	Parsing Error: The AUTHENTICATION specification which is required for each request is missing.
00E	ING_timeout_missing	Parsing Error: The required TIMEOUT parameter is missing in the request.
00F	ING_OCFCMD_truncated	Parsing Error: The HW function (OCFCMD) exceeds the maximum allowed length, which is 40 characters.
010	ING_OCFCMD_missing	Parsing Error: No HW function (OCFCMD) was specified in the request.
020	ING_SNMP_noIP_address	Parameter Resolution Error: The SNMP interface was specified for the HW request, but no IP address information is available.
021	ING_OCF_resolve_failed	Parameter Resolution Error: No HW function name to resolve, same as error 010.

Reason Code	Error String	Error Description
022	ING_OCF_not_resolved	Parameter Resolution Error: An invalid HW function name was detected.
030	ING_nt_alloc_error	Storage Allocation Error: The Netid base table could not be allocated using CNMNAMS services.
031	ING_img_alloc_error	Storage Allocation Error: The storage for a system image could not be allocated using CNMNAMS services.
032	ING_img_locate_error	Storage Allocation Error: The previously allocated storage for a system image could not be located using CNMNAMS services.
033	ING_notinit_error	Storage Allocation Error: A HW function request was issued for a processor / system image without having allocated storage for that processor / system image. This happens if no INITCOM request was made prior the first a HW function request.
050	ING_notinitialized_error	HW Function Error: A HW function request was issued for a processor / system image without having done an INITCOM. Same as error 033.
051	ING_imgnotfound_error	HW Function Error: A HW function request was issued for a system image which could not be located as an image belonging to the addressed CPC.
052	ING_funcunknown_error	HW Function Error: A unknown HW function name requested. Same as error 022.
053	ING_nocpcobject_error	HW Function Error: GETCLUSTER failed. Cluster list attribute not resolved by the processor support element.
054	ING_nocluster_error	HW Function Error: GETCLUSTER failed. The cluster list returned by the processor support element was empty.
055	ING_nohwstatus_error	HW Function Error: A HW function which requires to determine the status of the object prior execution cannot be processed because the object status cannot be determined. This error is valid only for processors where the FORCE option has to be emulated by INGHWCOM.
056	ING_disruptive_cmd	HW Function Error: A disruptive HW function was requested without the FORCE option and the processor / image object is in an operational state. INGHWCOM uses FORCE(NO) (allow no disruptive commands) as default. If you want to allow disruptive commands you must specify the FORCE option in the INGHWCMD request.
057	ING_noistatus_error	HW Function Error: CBU failed. a CBU function was requested but the current CBU status cannot be determined.
058	ING_noiobject_error	HW Function Error: CBU failed. The processor hardware does not support the CBU installed object attribute.
059	ING_cbustatus_error	HW Function Error: CBU failed. A CBU status was returned not allowing the request.
0A0	ING_invalid_task	HW Task Error: The HW communication interface is running on a NetView task which is not the configured task. Module INGHWCOM terminates. Verify that INGRCUST contains a valid autotask name for the HWOPER02 and HWOPER01 keywords (msys only). For System Automation, make sure you have defined autotask names for the keywords HWOPER01 and HWOPER02 in your active automation policy. Use the SA Dialog to verify this.

Reason Code	Error String	Error Description
0A2	ING_config_error	HW Task Error: The configuration information about the NetView autotask names to be used for the HW communication interface cannot be retrieved. This happens if the interface is called but the msys for Ops / System Automation initialization is not complete. This error also happens if the autotasks are not defined. See error code 0A0 for additional information.

SNMP Data Exchange Services "0B100xxx"

Note, that this set of condition codes applies to SNMP connections only

The following table lists the condition codes returned in an error case of the following INGHWCMD functions: INITCOM, TERMCOM, ACTIVATE, DEACTIVATE, SYSRESET, START, STOP, RESTART, LOAD, CBU, EXTERNAL, GETSSTAT, GETISTAT.

The condition code data "xxx" prefixed by 0B100 is returned as part of the AOFA0001, AOFA0002, AOFA0004, AOFA0017, AOFA0018 response message with a status value of REJECTED or FAILED.

The table data is copied from "zSeries 900 Application Programming Interface" hardware documentation.

001	HWMCA_DE_NO_SUCH_OBJECT
002	HWMCA_DE_INVALID_DATA_TYPE
003	HWMCA_DE_INVALID_DATA_LENGTH
004	HWMCA_DE_INVALID_DATA_PTR
005	HWMCA_DE_INVALID_DATA_VALUE
006	HWMCA_DE_INVALID_INIT_PTR
007	HWMCA_DE_INVALID_ID_PTR
008	HWMCA_DE_INVALID_BUF_PTR
009	HWMCA_DE_INVALID_BUF_SIZE
010	HWMCA_DE_INVALID_DATATYPE_PTR
011	HWMCA_DE_INVALID_TARGET
012	HWMCA_DE_INVALID_EVENT_MASK
013	HWMCA_DE_INVALID_PARAMETER
014	HWMCA_DE_READ_ONLY_OBJECT
015	HWMCA_DE_SNMP_INIT_ERROR
	This is a retryable condition code.
016	HWMCA_DE_INVALID_OBJECT_ID
017	HWMCA_DE_REQUEST_ALLOC_ERROR
018	HWMCA_DE_REQUEST_SEND_ERROR

019	HWMCA_DE_TIMEOUT
020	HWMCA_DE_REQUEST_RECV_ERROR
021	HWMCA_DE_SNMP_ERROR Check if the SNMP API is enabled on the SE.
022	HWMCA_DE_INVALID_TIMEOUT
028	HWMCA_DE_INVALID_HOST
029	HWMCA_DE_INVALID_COMMUNITY
099	HWMCA_DE_TRANSPORT_ERROR

SNMP Command Services "0B200xxx"

Note, that this set of condition codes applies to SNMP connections only

The following table lists the condition codes returned in an error case of the following hardware functions: ACTIVATE, DEACTIVATE, SYSRESET, START, STOP, RESTART, LOAD, CBU, EXTERNAL.

The condition code data "xxx" prefixed by 0B200 is returned as part of the AOFA0001 response message with a status value of REJECTED or FAILED.

The table data is copied from "zSeries 900 Application Programming Interface" hardware documentation.

001	HWMCA_CMD_NO_SUCH_OBJECT
002	HWMCA_CMD_INVALID_DATA_TYPE
003	HWMCA_CMD_INVALID_DATA_LENGTH
004	HWMCA_CMD_INVALID_DATA_PTR
005	HWMCA_CMD_INVALID_DATA_VALUE
006	HWMCA_CMD_INVALID_INIT_PTR
007	HWMCA_CMD_INVALID_ID_PTR
010	HWMCA_CMD_INVALID_DATATYPE_PTR
013	HWMCA_CMD_INVALID_PARAMETER
017	HWMCA_CMD_REQUEST_ALLOC_ERROR
018	HWMCA_CMD_REQUEST_SEND_ERROR
019	HWMCA_CMD_TIMEOUT
020	HWMCA_CMD_REQUEST_RECV_ERROR
021	HWMCA_CMD_SNMP_ERROR
022	HWMCA_CMD_INVALID_TIMEOUT
023	HWMCA_CMD_INVALID_CMD
024	HWMCA_CMD_OBJECT_BUSY
025	HWMCA_CMD_INVALID_OBJECT

026	HWMCA_CMD_COMMAND_FAILED
027	HWMCA_CMD_INITTERM_OK
028	HWMCA_CMD_CBU_DISRUPTIVE_OK
029	HWMCA_CMD_CBU_PARTIAL_HW
030	HWMCA_CMD_CBU_NO_SPARES
031	HWMCA_CMD_CBU_TEMPORARY
032	HWMCA_CMD_CBU_NOT_ENABLED
033	HWMCA_CMD_CBU_NOT_AUTHORIZED
034	HWMCA_CMD_CBU_FAILED
035	HWMCA_CMD_CBU_ALREADY_ACTIVE

BCP Internal Interface Transport Services "0Bx00xxx"

Note, that this set of condition codes applies to BCP internal interface connections only.

The following table lists the condition codes returned in an error case of the following INGHWCMD functions: INITCOM, TERMCMD, ACTIVATE, DEACTIVATE, SYSRESET, START, STOP, RESTART, LOAD, CBU, EXTERNAL, GETSSTAT, GETISTAT.

The condition code data "xxx" prefixed by 0B100 or 0B200 is returned as part of the AOFA0001, AOFA0002, AOFA0004, AOFA0017, AOFA0018 response message with a status value of REJECTED or FAILED.

Reason Code	Error Description
100	A problem was encountered prior to sending the request to the HSAET32 API for processing. This is likely due to a failure to an environmental error. Check if the Support Element is fully operational. A running reboot of the SE may caused this problem.
101	A problem was encountered prior to sending the request to the HSAET32 API for processing. This is likely due to a failure to properly enable the API, however may be due to parameter and/or environmental errors.
102	A report list overflow occurred. This return code should not currently be issued for SNMP requests, however is included for OCF query (Query-Read-Cluster) compatibility.
110	The issuer of the request is not (RACF) authorized to the requested function. Note that (like HCD) the HSAET32 services require that RACF or a compatible SAF product be installed and operational.
111	The control block ID or version of the HSDB passed to HSAET32 services is invalid. For hwmcaapi requests, this indicates that HSAPHCPI is incompatible with the supporting HSAPHARI module.
112	The requested function is invalid or not supported by the current level of HSAET32 services. For hwmcaapi requests, this indicates an incompatibility between HSAPHCPI and the supporting HSAPHARI module.
113	The control block ID of the request list passed to HSAET32 services is invalid or inappropriate for the requested function. For hwmcaapi requests, this indicates a problem in module HSAPHCPI.
114	The request list entry count passed to HSAET32 services is invalid or inappropriate for the requested function. For hwmcaapi requests, this indicates a problem in module HSAPHCPI.

Reason Code	Error Description
115	The request list entry pointer passed to HSAET32 services is null and therefore invalid. For hwmcaapi requests, this indicates a problem in module HSAPHCPI.
116	Some of the input areas passed to HSAET32 services exist in a storage area that the caller does not have authority to fetch/update.
117	The input parameter list generated by the HSAXHARI (or CBDIHSD) macro does not have the correct version ID or type, or does not point to an HSDB.
118	The control block ID of the Output Report request list passed to HSAET32 services is invalid or inappropriate for the requested function. For hwmcaapi requests, this indicates a problem in module HSAPHCPI.
119	The session token is invalid. This is probably due to a previous failure of the hwmcaterminate request being issued for the session, or improper modification of the HWMCA_SCLP_TARGET_INFO structure.
120	The host environment does not support HSAET32 services. HSAET32 services are not currently on VM hosts.
121	An address space resource manager could not be established.
122	A task resource manager could not be established.
123	The HSAET32 associated recovery routine (HSAPHARR) was entered due to an unexpected error processing the request.
124	The CBDMHWA CSECT could not be found in the nucleus.
125	The HSAET32 monitor exit (HSAPHMON) could not be established as the secondary ET32 listner exit for the application.
126	The system date and time could not be obtained to correlate HRE and associated MDS-MU's.
127	A failure attempting occurred attempting to access the HWAX.
129	An attempt to send the MDS_MU requests across the BCP internal interface interface failed.
130	Either the EP_OPERATIONS_MGMT vector (9F22) from the event type 30 data was not available or its length was invalid.
131	Either the application name-group for the EP_OPERATIONS_MGMT application (event type 30 data) was not returned in the 9F22 vector or its length was invalid.
132	Either the NetID of the local support element was not returned in the application name-group for the EP_OPERATIONS_MGMT application (event type 30 data) or its length was invalid.
133	Either the NAU of the local support element was not returned in the application name-group for the EP_OPERATIONS_MGMT application (event type 30 data) or its length was invalid.
134	Either the CPC image name vector (9F70) from the event type 30 data was not available or its length was invalid.
136	Either the primary OCF name vector (9F81) from the event type 30 data was not available or its length was invalid.
150	HSAPHSPI was unable to establish an ESTAEX recovery environment.
151	HSAPHSPI identified a parameter that is not contextually valid.
152	HSAPHSPI identified a missing parameter that is contextually required.
153	HSAPHSPI identified a parameter value that is syntactically incorrect.
204	The request was accepted by the local support element and will be processed asynchronously. No further reason code is provided. (This function is not currently used by the hwmcaapi implementation).
208	Execution of request was failed by the target support element. This indication is normally accompanied by a condition report which is returned as the error reason, and may also be accompanied by sense data further identifying the cause of the failure.

Reason Code	Error Description
212	The request was rejected by the local support element. This indication is normally accompanied by a condition report which is returned as the error reason.
216	An MDS-MU error message was received from the target support element. The condition report code is returned as the error reason.
220	HSAPHMON detected an structural error while processing the incoming report from the target support element. An internal reason code is generated to further describe the request.
224	No response was received from the target support element within the time interval designated for the request. No further reason code provided.
228	An error was detected in a request list entry. An internal reason code is generated to identify the field in error. (This code is not used for hwmcaapi requests).
232	A routing error has occurred while forwarding the requests for processing. This indication is normally accompanied by a condition report which is returned as the error reason.

Appendix E. Sense Codes, Hardware Object Status Summary

Sense Codes

Note, that for BCP internal interface connections the sense codes are copied from the request response report information into the AOFAxxxx messages. For SNMP connections, the sense data is taken from the HWMCA_EVENT_COMMAND_RESPONSE return codes, which is currently not supported. For SNA connections this information is provided in the CSAAXxxx messages, which can be found in the netlog.

The following table lists the sense codes returned in an error case of the following hardware functions: ACTIVATE, DEACTIVATE, SYSRESET, START, STOP, RESTART, LOAD, CBU, EXTERNAL.

The table data is copied from "Managing Your Processors" hardware documentation. Note, that the data listed here as sense codes correspond to the Operations Management Condition Code Reference in the above named documentation. For BCP internal interface and SNMP connections only a subset of this SNA set applies.

Sense codes are returned in the SENSE field of the AOFAxxxxx messages.

0806000A	RESOURCE UNKNOWN
-----------------	-------------------------

Explanation: The profile name (CNAME) specified in a operations command is not recognized by the receiving node.

System Programmer Response: Correct the configuration identifier and resend the request.

08090000	Mode inconsistency: The requested function cannot be performed in the present state of the receiver.
-----------------	---

Explanation: This command is prohibited because the target is in an incompatible mode. For example, an ITIMER request is not accepted when the system is power-on reset in LPAR mode.

System Programmer Response: This function cannot be performed in the present state of the receiver. Retry the request after the target mode status has changed.

08090001	Mode inconsistency: The requested function cannot be performed in the present state of the receiver.
-----------------	---

Explanation: Acceptance of the command is prohibited because the target is in an incompatible mode. For example, an ITIMER request is not accepted when the system is power-on reset in LPAR mode.

System Programmer Response: None. This function cannot be performed in the present state of the receiver.

08090027	Mode inconsistency: The requested function cannot be performed in the present state of the receiver.
-----------------	---

Explanation: The receiving Hardware Management Console is not in the correct state to automatically dial out using the attached modem.

System Programmer Response: Ensure the receiving Hardware Management Console is customized to use the auto-dial and RSF functions.

08090051	Mode inconsistency: The requested function cannot be performed in the present state of the receiver.
-----------------	---

Explanation: Operations management control is not enabled.

System Programmer Response: Enable the system for automated operations and resend the request. Ensure that the Emergency Power Off switch is on.

080A000A	Permission rejected: The receiver has denied an implicit or explicit request of the sender.
-----------------	--

Explanation: A STATLEV request was rejected because it was not compatible with the status reporting values set in the receiver.

System Programmer Response: Correct the STATLEV value and resend the request.

080A000C	Permission rejected: The receiver has denied an implicit or explicit request of the sender.	08120012	Insufficient resource: The receiver cannot act on the request because of a temporary lack of resource.
Explanation: A SETCLOCK request has failed because it required that a clock be set in a configuration where a dominant timing source has priority.		Explanation: A timed command was rejected because the OCF timed operations queue was full.	
System Programmer Response: If the Sysplex Timer is the dominant timing source, then the SOURCE, TIME, UTCO, and OFFSET operands cannot be used in the command string. Remove these operands and resend the request.		System Programmer Response: Cancel any unnecessary scheduled requests and resend the command.	
080C0005	Procedure not supported: A procedure specified is not supported in the receiver.	08150001	Function active: A request to activate an element or procedure was received, but the element or procedure was already active.
Explanation: The command is not supported.		Explanation: Unable to perform the command because the target CPC Subset or CPC Image is operational and the force operand has not indicated the override selection.	
System Programmer Response: Resend the request using a supported command, if possible.		System Programmer Response: Put the system in the appropriate state and resend the command.	
080C0007	Procedure not supported: A procedure specified is not supported in the receiver.	081A0000	Request sequence error.
Explanation: A request for a function is supported by the receiver, but the resource identified in the request does not support that function. This function cannot be canceled.		Explanation: Unable to perform the command because the target partition is in the deactivated state.	
System Programmer Response: None.		System Programmer Response: Activate the logical partition, then resend the original request.	
080F0001	End-user not authorized: The requesting end-user does not have access to the requested resource.	081A0009	Request sequence error.
Explanation: Authorization checks have not been successfully passed.		Explanation: Unable to perform command because power is not on.	
System Programmer Response: Correct the command authorization-token and resend the request.		System Programmer Response: Send a POWERON or ACTIVATE command, then resend the original request.	
08120000	Insufficient resource: The receiver cannot act on the request because of a temporary lack of resource.	081A000A	Request sequence error.
Explanation: System resources are temporarily busy.		Explanation: Unable to perform command because power-on reset is not complete.	
System Programmer Response: Resend command if required.		System Programmer Response: Send a POWERON or ACTIVATE command, then resend the original request.	
08120011	Insufficient resource: The receiver cannot act on the request because of a temporary lack of resource.	081A000B	Request sequence error.
Explanation: Insufficient storage is available to the target component to satisfy the request.		Explanation: Unable to perform command because the targeted CP is not in the stopped state.	
System Programmer Response: Resend command.		System Programmer Response: Send a STOP command, then resend the original request.	
		081A000E	Request sequence error.
		Explanation: Unable to perform command because the interval timer is present only when the CPC Image is operating in S/370 mode.	
		System Programmer Response: None. The requested command cannot be performed when the system is	

power-on reset in either ESA/390 mode or LPAR mode.			
081A0010	Request sequence error.	081C00BA	Request not executable: The requested function cannot be executed because of a permanent error condition in the receiver.
Explanation: The request is rejected or failed because the target resource is already in the state or condition that the request would have provided.		Explanation: The receiver has an error resulting from a licensed internal code problem that prevents execution of the request.	
System Programmer Response: None. The requested command has already been performed.		System Programmer Response: Retry the operation. If the problem persists, follow local procedures for reporting a processor complex problem.	
081C0005	Request not executable: The requested function cannot be executed because of a permanent error condition in the receiver.	081F0000	Request was canceled by an operator.
Explanation: A power-on request failed.		Explanation: The operator has canceled the requested function.	
System Programmer Response: Verify power is available and resend the command.		System Programmer Response: Resend the command if required.	
081C0006	Request not executable: The requested function cannot be executed because of a permanent error condition in the receiver.	082D0001	Busy.
Explanation: A POR(YES) or POR(IML) failed. This may be accompanied by a hardware alert.		Explanation: Resources needed to process the request are being used.	
System Programmer Response: Retry the operation. If the problem persists, follow local procedures for reporting a processor complex problem.		System Programmer Response: Wait for the resources to be released, then resend the request.	
081C0007	Request not executable: The requested function cannot be executed because of a permanent error condition in the receiver.	08380000	Request not executable because of resource or component state incompatibility: The request is not executable because it is not compatible with the state of a resource or component in the receiver.
Explanation: An operating system load request (for example, LOAD) failed.		Explanation: Unable to perform the command because the system is in an invalid state.	
System Programmer Response: Retry the operation. If the problem persists, follow local procedures for reporting a processor complex problem.		System Programmer Response: Put the system in a state that is compatible with the requested command and resend the request.	
081C000A	Request not executable: A POWEROFF request cannot be performed because of a permanent error condition in the receiver.	08380017	Request not executable because of resource or component state incompatibility: The request is not executable because it is not compatible with the state of a resource or component in the receiver.
Explanation: A power off request failed due to an unexpected power status.		Explanation: Execution of the request referred to in a cancel command has proceeded too far to cancel.	
System Programmer Response: Reset any abnormal power conditions at the receiver, such as tripped circuit breakers, and retry the power off command. If the problem persists, follow local procedures for reporting a processor complex problem.		System Programmer Response: None. The request you want to cancel is already being processed.	

08380018	Request not executable because of resource or component state incompatibility: The request is not executable because it is not compatible with the state of a resource or component in the receiver.	08380038	Request not executable because of resource or component state incompatibility: The request is not executable because it is not compatible with the state of a resource or component in the receiver.
Explanation: Cancellation of the request referred to in a cancel command cannot be done without disrupting CPC Subset resources.		Explanation: Request will not be honored because it requires that the resource not exist, but the resource already exists.	
System Programmer Response: None. The request failed during processing and cannot be completet.		System Programmer Response: If the resource is a profile name, change the profile name in the command string and resend the request.	
08380019	Request not executable because of resource or component state incompatibility: The request is not executable because it is not compatible with the state of a resource or component in the receiver.	084C0001	Permanent insufficient resource: Receiver cannot act on the request because resources required to honor the request are permanently unavailable. The sender should not retry immediately because the situation is not transient.
Explanation: The timing-window time specified in a command request has expired. The request will not be honored.		Explanation: Disk space is unavailable to complete the request.	
System Programmer Response: Resend the commabnd with a valid timing-window.		System Programmer Response: Follow local procedures for reporting a processor complex problem.	
0838001B	Request not executable because of resource or component state incompatibility: The request is not executable because it is not compatible with the state of a resource or component in the receiver.	084F0000	Resource not available: A requested resource is not available to service the given request.
Explanation: Request will not be honored because it was submitted to a node at a time when a local operator or other application reserved control of the node.		Explanation: A resource error exits which may indicate a configuration problem or insufficient resource to execute the command.	
System Programmer Response: Request the local operator to release control (log off), or retry later.		System Programmer Response: Check the accompanying SDATA codes, if any exist, to determine the specific resource error.	
08380037	Operating system is not receiving. The request is not executable because the operating system is not able to respond because it is in an inactive or quiesced state.	085B0000	Unknown resource name: The identified resource, required to complete the requested command, is not known.
Explanation: Request will not be honored because it requires that the resource operating system is in an active state.		Explanation: The profile name specified in the AUTOACT operand of the RESET profile is not recognized by the receiving node.	
System Programmer Response:		System Programmer Response: Correct the profile name and resend the request.	
		085B0003	Unknown resource name: The identified resource, required to complete the requested commands, is not known.
		Explanation: The clock identifier specified in a SETCLOCK command is in error. Either the OCF is unknown to the receiver, or the TOD is invalid.	
		System Programmer Response: Correct the CLOCK parameter and resend the request.	

085B0004	Unknown resource name: The identified resource, required to complete the requested command, is not known.	085C0002	System exception: The node experiences an exception condition within a resident system or subsystem that inhibits further processing by the component.
Explanation:	The timing source name specified in a SETCLOCK command is unknown to the receiver.	Explanation:	If accompanied by an SDATA code, this condition code indicates that a complete activation failed. Activation was completed only through the power-on reset step. See the SDATA code for additional information.
System Programmer Response:	Correct the SOURCE operand and resend the request.		
085B0005	Unknown resource name: The identified resource, required to complete the requested command, is not known.		
Explanation:	The correlator referred to by a cancel command is unknown to the receiver, or represents a command already completed.		If there is no SDATA code, the exception is identified as a permanent system-related problem. This may be accompanied by a hardware alert.
System Programmer Response:	None. There is no pending request to cancel.	System Programmer Response:	If the code is returned for an ACTIVATE request, to complete activation, send another ACTIVATE request to complete the initial program load.
085B0006	Unknown resource name: The identified resource, required to complete the request command, is not known.		For all other requests, retry the operation. If the problem persists, follow local procedures for reporting a processor complex problem.
Explanation:	The timing source name specified in a SETCLOCK command is the same as the clock name to be set.	086B0B10	Sub-field value invalid.
System Programmer Response:	Correct either the CLOCK or the SOURCE operand and resend the request.	Explanation:	An invalid XATIME operand was specified on the command request.
085C0000	System exception. The node experiences an exception condition within a resident system or subsystem that inhibits further processing by the component.	System Programmer Response:	Correct the XATIME operand and resend the request.
Explanation:	An internal error has occurred with the processing of this request. This may be accompanied by a hardware alert.	086B0B20	Sub-field value invalid.
System Programmer Response:	Retry the operation. If the problem persists, follow local procedures for reporting a processor complex problem.	Explanation:	An invalid XBTIME operand was specified on the command request.
085C0001	System exception: The node experiences an exception condition within a resident system or subsystem that inhibits further processing by the component.	System Programmer Response:	Correct the XBTIME operand and resend the request.
Explanation:	The exception is identifiable as a system-related problem. This may be accompanied by a hardware alert.	086B0B30	Sub-field value invalid.
System Programmer Response:	Retry the operation. If the problem persists, follow local procedures for reporting a processor complex problem.	Explanation:	An invalid INTERVAL operand was specified on the command request.
		System Programmer Response:	Correct the INTERVAL operand and resend the request.
		086B0B40	Sub-field value invalid.
		Explanation:	An invalid COUNT operand was specified on the command request.
		System Programmer Response:	Correct the COUNT operand and resend the request.
		086B8110	Sub-field value invalid.
		Explanation:	If returned for a CANCEL request, the OCF does not support the canceling of this CANCEL TCORR(hex-value) request.
			For all other requests, this code indicates that an

invalid FORCE operand was specified on the command request.S

System Programmer Response: None, if the code is returned for a CANCEL request; this function cannot be performed. For all other requests, correct the FORCE operand and resend the request.

086B8115 Sub-field value invalid.

Explanation: An invalid CLOCK operand was specified on a SETCLOCK command.

System Programmer Response: Correct the operand and resend the request.

086B8120 Sub-field value invalid.

Explanation: If returned on a SETCLOCK command, an invalid SOURCE operand was specified. If returned on an ACTIVATE command, an invalid CHANGE operand was specified.

System Programmer Response: Correct the operand and resend the request.

086B8130 Sub-field value invalid.

Explanation: If returned for a CANCEL request, the OCF does not support the canceling of this CANCEL DEFERRED request.

If returned for a SETCLOCK, this code indicates that an invalid TIME operand was specified on the request.

System Programmer Response: None, if the code is returned for a CANCEL request; this function cannot be performed. If the code is returned for a SETCLOCK request, correct the TIME operand and resend the request.

086B8140 Sub-field value invalid.

Explanation: An invalid UTCO operand was specified on a SETCLOCK request.

System Programmer Response: Correct the operand and resend the request.

086B8150 Sub-field value invalid.

Explanation: An invalid UTCO operand was specified on a SETCLOCK request.

System Programmer Response: Correct the operand and resend the request.

086D8115 Required sub-field missing.

Explanation: The CLOCK operand is missing on a SETCLOCK command.

System Programmer Response: Correct the operand and resend the request.

08A80001 Multiple-domain support routing exception: the MDS router in the reporting NAU is unable to perform the required routing for an MDS MU.

Explanation: Origin or destination NAU name unknown. For NetView, either the origin or the destination Net ID in the MDS header is invalid.

System Programmer Response: Ensure the node is available and correct the configuration definition.

08A80002 Multiple-domain support routing exception: The MDS router in the reporting NAU is unable to perform the required routing for an MDS MU. MS application name not recognized.

Explanation: The NAU directory services are unavailable.

System Programmer Response: Retry the operation after the directory service problem has been corrected.

08A80003 Multiple-domain support routing exception: the MDS router in the reporting NAU is unable to perform the required routing for an MDS MU. MS application name not recognized.

Explanation: For applications using the NetView LU 6.2 transport, this sense code indicates that one of the MS applications named in the request is not registered with NetView.

System Programmer Response: Ensure the node is available and retry the operation.

08A80009 Multiple Domain Support (MDS) routing exception: The MDS router in the reporting NAU is unable to perform the required routing for a MDS MU.

Explanation: Destination not supported by the reported network node (NN). The NN has received an MDS MU from another node that cannot be routed.

System Programmer Response: Ensure the configuration is correct and the node is available, then retry the operation.

08A8000A Multiple-domain support routing exception: The MDS router in the reporting NAU is unable to perform the required routing for an MDS MU. Unrecoverable session failure. Unrecoverable TP failure in remote node.

Explanation: The MDS_Send TP in the reporting node was unable to send the message because of an allocation error. Retries have been exhausted.

System Programmer Response: Resend the operation after the network node becomes available.			
08A8000B	Multiple-domain support routing exception: the MDS router in the reporting NAU is unable to perform the required routing for an MDS MU. The network had a failure during an operation request transmission; Unrecoverable TP failure.	08A90001	Multiple-domain support transaction failure: The reporting MDS router or MS application has detected a condition that has impacted an outstanding unit of work (identified by the unit-of-work correlator of the MDS error message) or MDS MU.
Explanation: The MDS_Send TP in the reporting node was unable to send the message because of a TP failure in a remote node. Retries have been exhausted.		Explanation: The failure was caused by an outage of a CPSVCMG session.	
System Programmer Response: After the network problem has been corrected, retry the operation request.		System Programmer Response: Ensure the node is available and retry the operation.	
08A8000F	Multiple-domain support routing exception: the MDS router in the reporting NAU is unable to perform the required routing for an MDS MU; MS application congestion.	08A90003	Multiple-domain support transaction failure: The reporting MDS router or MS application has detected a condition that has impacted an outstanding unit of work (identified by the unit-of-work correlator of the MDS error message) or MDS MU; Unit of work canceled by reporting MS application program.
Explanation: The MDS router in the destination NAU is unable to communicate with the destination MS application because of local congestion (implementation buffer space for queuing additional MDS MU(s) has been exhausted).		Explanation: The unit of work has been canceled because of a timeout in the reporting MS application program. For applications using the NetView LU 6.2 transport, the timeout value is determined by the SECONDS parameter on the transport send service, of the RCVREPLY value set by the DEFAULTS command.	
System Programmer Response: Resend the operation request after the congestion has cleared.		System Programmer Response: Retry the operation.	
08A80013	Multiple-domain support routing exception: the MDS router in the reporting NAU is unable to perform the required routing for an MDS MU; Session outage involving an MDS LU 6.2 session has occurred. For NetView, further sense data is contained within the supplemental report subfield in the SNA condition report containing this sense code.	08A90004	Multiple-domain support transaction failure: The reporting MDS router or MS application has detected a condition that has impacted an outstanding unit of work (identified by the unit-of-work correlator of the MDS error message) or MDS MU; Unit of work canceled by reporting MDS router.
Explanation: The last session to the indicated destination has been deactivated. For NetView, further sense data is contained within the supplemental report subfield in the SNA condition report containing this sense code.		Explanation: The unit of work has been canceled by a garbage-collection time-out in the reporting MDS router. For applications using the NetView LU 6.2 transport, this sense code is returned if the timeout value for a request matches that set by MAXREPLY on the DEFAULTS command.	
System Programmer Response: Retry the operation after the destination session is reinstated.		System Programmer Response: Retry the operation. If the problem persists, follow local procedures for reporting a processor complex problem.	
		08B20000	A timeout occurred trying to issue the command. The data transmission between the application in the processor and the support element was not completed, the command acceptance report was not received before the timeout condition occurred.
		Explanation: A timeout has occurred while waiting for transmission of data between two applications.	

System Programmer Response: Retry the operation. If the problem persists, follow local procedures for reporting a processor complex problem.

08B20002 Data transmission failure: The data transmission between an application in the support element and an application in the processor was incomplete causing abnormal termination of the function.

Explanation: A timeout has occurred while waiting for transmission of data between two applications.

System Programmer Response: Retry the operation. If the problem persists, follow local procedures for reporting a processor complex problem.

1003000D Function not supported.

Explanation: The function identified in the request is not supported.

System Programmer Response: Correct the command string and resend the request.

100B0001 Required structure absent.

Explanation: An operand required by the command was not found in the command string.

System Programmer Response: Enter the required operand and resend the request.

100B0002 Precluded structure present.

Explanation: The operand indicated in SDATA is not allowed by the command or by other operands.

System Programmer Response: Remove the precluded operand or correct the command and resend the request.

100B0003 Multiple occurrences of a non-repeatable structure.

Explanation: A value that cannot be repeated was detected in the command string.

System Programmer Response: Change the duplicate value(s) to unique value(s) and resend the request.

100B0004 Excess occurrences of a repeatable structure.

Explanation: A value that can be repeated has exceeded the maximum number of occurrences permitted in the command string.

System Programmer Response: Reduce the number of occurrences of the value in error to an acceptable number and resend the request.

100B0006 Length outside specified range.

Explanation: The length of the operand indicated in SDATA is outside the allowable range.

System Programmer Response: Correct the operand data-value and resend the request.

100B0007 Length exception; length arithmetic is out of balance.

Explanation: The length of the operand value for the operand indicated in SDATA is not consistent with the allowable values.

System Programmer Response: Correct the operand data-value and resend the request.

100B000A Required compination of structures and data values absent.

Explanation: One command operand or data-value that is present in the command string requires another operand or data-value that is absent.

System Programmer Response: Correct the operand or data-value and resend the request.

100B000B Precluded combination of structures and data values present.

Explanation: One command operand or data-value is in conflict with one or more other operands or data-values.

System Programmer Response: Remove the precluded operand(s) or correct the command and resend the request. Also check the activation profile(s) used for activation, as the error may be the result of incorrect profile data.

100B000C Unknown or unsupported data-value.

Explanation: The data value in the operand indicated by SDATA is either unknown or unsupported.

System Programmer Response: Correct the operand data-value and resend the request.

100B000D Incompatible data-values.

Explanation: The data value in the operand indicated by SDATA is not compatible with this or other values.

System Programmer Response: Correct the conflicting operand data-value and resend the request.

100B000E Precluded character present.

Explanation: The data value in the operand indicated by SDATA contains a character that is not allowed for the indicated operand.

System Programmer Response: Correct the operand

data-value and resend the request.

100B000F Data-value out of range.

Explanation: The data value in the operand indicated by SDATA is not within the range allowed for the indicated operand.

System Programmer Response: Correct the operand data-value and resend the request.

100B0011 Precluded data-value.

Explanation: The data value in the operand indicated by SDATA is not allowed by this or other operands.

System Programmer Response: Replace the precluded operand data-value and resend the request.

100B0012 Recognized but unsupported structure.

Explanation: The operand indicated by SDATA is recognized but not supported by the target support element.

System Programmer Response: Remove the

unsupported operand and resend the request.

80180002 Resource unknown

Explanation: The secondary OCF specified in the OCFNAME operand is not recognized, or an invalid command was issued to the OCF. For scheduled operations, this may indicate that the logical partition existed when the command was validated and placed on the timed operations queue, but was no longer valid when the timing-window for processing arrived.

System Programmer Response: Ensure the system is power-on reset in LPAR mode and the secondary-name in the OCFNAME operand matches a logical partition name in the active IOCDS, or specify the correct OCFNAME in the command.

80180003 Resource unknown.

Explanation: Unrecognized Destination Application Name in the DAN operand of RUNCMD.

System Programmer Response: Correct the operand data-value and resend the request.

Hardware Object Status Summary

The following lists the status values for CPC and image objects provided by the z900 API. The status description was taken from the HMC online help, since the API documentation does not provide this information. Note, that the z900 SNMP API provides only subset of the object states provided by the HMC.

OPERATING	Image: All of the image's processors are operating. CPC: All of the CPC's processors are operating.
NO POWER	CPC: CPC power is off.
NOT OPERATING	Image: None of the image's processors are operating, but the exact status of the processors vary. CPC: <u>If a power-on reset has not been performed:</u> The CPC's processors cannot operate until a power-on reset of the CPC is performed. <u>If a power-on reset was performed:</u> None of the CPC's processors are operating, but the exact status of the processors vary.
NOT ACTIVATED	Image: The image is defined in the CPC's current input/output (I/O) configuration, but is not activated.
EXCEPTIONS	Image: At least one of the image's processors is operating, and at least one processor is not operating, but the exact status of the processors vary. CPC: At least one of the CPC's processors is operating, and at least one processor is not operating, but the exact status of the processors vary.
STATUS CHECK	Image: The CPC is not communicating with the support element. The status of the image and its CPs cannot be determined. CPC: The CPC is not communicating with the support element.

POWERSAVE	<p>CPC: Utility power for the CPC failed, and one or more of its active control programs put the CPC in a power save state. The CPC is using only enough power from its alternate, temporary power source to preserve data for the control programs that put it in the power save state.</p> <p>Image: The image cannot operate until power for the CPC is restored.</p>
LINK NOT ACTIVE	<p>CPC: The CPC's support element is not communicating with this HMC. The status of the CPC cannot be determined.</p>
SERVICE	<p>CPC: A console operator enabled service status for the CPC (ordinarily done at the request of a service representative to allow providing service for the CPC).</p>
SERVICE REQUIRED	<p>No explanation found on the HMCs.</p>

Glossary

This glossary defines technical terms and abbreviations used in z/OS Managed System Infrastructure for Operations documentation. If you do not find the term you are looking for, view *IBM Glossary of Computing Terms*, or *Tivoli Glossary* located at:

<http://www.ibm.com/ibm/terminology>

<http://www.tivoli.com/support/documents/glossary>

A

authorized (message) receiver. The operator who is authorized to receive all the unsolicited and authorized messages that are not routed to a specific operator.

automation router task. The automation router task receives the unsolicited messages that are defined as automatable and initiates the automated response if the automation table contains an entry for this message.

autotask. An unattended operator station task that does not require a terminal or a logged-on user. Autotasks can run independently of VTAM and are typically used for automated console operations.

C

CDS. Couple Data Set.

CF. Coupling Facility.

CFRM. Coupling facility resource management.

CFRM policy. A set of *coupling facility* and *structure* definitions. The purpose of a CFRM policy is to specify which coupling facilities and structures are available in the sysplex. There can be several CFRM policies, but only one of these can be active at any given time. Activating a different CFRM policy is possible at runtime. The CFRM policies are contained in the CFRM *couple data set*.

connector. An application (component) that is connected to a *structure*.

couple data set. A data set that contains control information about the sysplex or a sysplex-related function such as CFRM. Every MVS system of a sysplex must have access to the couple data set of all functions that are implemented in the sysplex.

coupling facility. A logical partition that provides storage for data exchange between applications across

the sysplex. The storage of a coupling facility is divided into areas that are called *structures*. Structures are identified by their name.

cross-system coupling facility (XCF). A component of MVS that supports cooperation between authorized programs running within a sysplex on the same or different members.

cross-system extended services (XES). A set of services that allow authorized applications or subsystems running in a sysplex to share data using a *coupling facility*.

D

data services task (DST). The msys for Operations subtask that gathers, records, and manages data in a VSAM file or a network device that contains network management information.

DST. Data Services Task.

duplexing. Maintaining two instances of a *structure* with the same name on different *coupling facilities* at the same time. Duplexing allows applications to maintain a backup version of their structures so that the structure is still available when one of its instances has failed.

E

EMCS. extended multiple console support.

L

logical partition. A logical processor complex within a physical processor complex that operates independently of the other logical partitions within that physical processor complex.

LPAR. Logically Partitioned Mode.

M

message automation table. A table that specifies the automated response to certain messages. The entries of the table determine which command is to be issued in response to the respective message, and which autotask is to execute the command.

message processing facility (MPF). An MVS facility that controls message display and message processing.

multisystem application. An application program that has various functions distributed across members of a sysplex.

O

operator station task (OST). The task that establishes and maintains the online session with the network operator. There is one operator station task for each network operator who logs on to msys for Operations.

P

policy. A set of sysplex-related definitions that is contained in a CDS and takes effect by being activated. Some CDSs can contain more than one policy. In this case only one policy can be active at a given time.

PPT. primary POI task.

primary POI task (PPT) . The subtask that processes all unsolicited messages that are received from the VTAM program operator interface (POI) and delivers them to the controlling operator or to the command processor.

preference list. An ordered list of *coupling facilities* that is associated with a *structure*. The preference list specifies the coupling facilities on which the structure can be allocated. Usually, the system allocates the structure on the first coupling facility in the list that is not excluded by other requirements (for example, the size of the structure). The preference list is part of the structure definition in a *CFRM policy*.

R

RACE. Resource Access Control Facility.

rebuild. A process by which data from an initial instance of a *structure* is reconstructed in another structure instance with the same name on the same or another *coupling facility*.

There are two types of rebuild, plain rebuild and *duplexing*. With the first type, the new structure instance will replace the initial one, whereas duplexing will simply create a second instance.

A rebuild can be accomplished by two methods, user-managed rebuild and (from release 8 onwards) system-managed rebuild. With user-managed rebuild, the *connectors* are responsible for reconstructing the data, whereas this is done by *XES* in case of a system-managed rebuild.

S

SFM. Sysplex Failure Management

SFM policy. A set of system-related entries that specify how a failure of that system or its connections within the sysplex is to be handled. There can be several SFM policies, but only one of these can be active at any given time. The active SFM policy can be switched at runtime. The SFM policies are contained in the *SFM couple data set*

SMF. System Management Facility.

solicited message. A message which is sent in response to an operator command, and which has a specific destination, such as an msys for Operations operator. Contrast *unsolicited message*

structure. A storage area in a *coupling facility* that is associated with a name. Structures serve to exchange data between applications components across the sysplex. They are allocated when an applications component requires to be connected to the structure; when all connected components have terminated the connection normally, the structure is deallocated again except when it has been declared persistent by a *connector*.

Most properties of the structure are specified by the connecting application, for instance its purpose and internal organization, whether the structure is persistent, and whether *rebuild* is allowed. However, the application does not select the coupling facility on which the structure is allocated; this is done by *XES* based on a *preference list*. The application only knows the name of the structure.

sysplex. One or more MVS systems that reside on one or more processors and are connected with each other so that programs in one of the systems can communicate with programs in another system. There are two types of sysplexes, basic and parallel. Parallel Sysplexes usually contain one or more *coupling facilities*.

sysplex failure management. An MVS component that manages system and connectivity failures within a sysplex according to a predefined policy. The SFM couple data set contains the active and eventual alternate policies.

system logger. An MVS component which enables applications to log data without needing to know how and where the data is stored, and which allows multisystem applications to use a common log.

System Management Facility (SMF). A standard feature of z/OS that collects and records a variety of system and job-related information.

T

task. (1) A process with certain properties that is defined to, and runs within, msys for Operations. (2)

Started task, an MVS procedure that is started with the START command of MVS. (3) A major unit in a processing sequence.

U

unsolicited message. A message that was not expected in response to an operator action. Contrast *solicited message*.

W

write-to-operator (WTO). A request to send a message to an operator at the z/OS operator console. This request is made by an application and is handled by the WTO processor, which is part of the z/OS supervisor program.

write-to-operator-with-reply (WTOR). A request to send a message to an operator at the z/OS operator console which requires a response from the operator. This request is made by an application and is handled by the WTO processor, which is part of the z/OS supervisor program.

WTO. Write-to-Operator.

WTOR. Write-to-Operator-with-Reply.

X

XCF. Cross-system coupling facility

XCF group. A set of program instances or functions across the sysplex that a multisystem application defines to XCF. Members of an XCF group can communicate with each other by XCF services.

XES. Cross-system extended services.

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
USA

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Deutschland
Informationssysteme GmbH
Department 3982

Pascalstrasse 100
70569 Stuttgart
Germany

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Programming interface information

This publication documents information that is NOT intended to be used as a Programming Interface of msys for Operations.

Trademarks

The following terms are trademarks or service marks of the IBM corporation in the United States or other countries or both:

IBM
MVS/ESA
NetView (Tivoli Systems Inc.)
OS/390
Parallel Sysplex
RACF
VTAM
z/OS

Microsoft, Windows, Windows NT and the Windows logo are registered trademarks of Microsoft Corporation.

UNIX is a registered trademark of the Open Group in the United States and other countries.

Other company, product, and service names may be trademarks or service marks of others.

Index

Special characters

%INCLUDE statement 199
&DOMAIN 33
&NV2I 34
&SQ1 34
&SQ2 34
&SYSCONE 25, 33
&VQ1 34
␣ symbol 35

A

ACF file manager command 147
active connector 14
ADDS, RACF command 64, 214
administration, system 159
ALL command 85, 93
alternate CDS 9
 allocating 128
 spare volumes 43, 128
 turn into primary CDS 9, 128
AOCTRACE command dialog 147
AOF532I message 69
AOF540I message 69
AOFUCST customization member 35, 155
 AUTO section 10, 11, 12, 13, 36
 CDS section 10, 43
 COMMON section 37
 ENQ section 40
 HW section 44
 IXC102A section 46
 PAGE section 38
 WTOBUF section 13, 47
asterisk (*) generic character 203
ATCCONxx VTAM configuration list 32
attributes, operator 192
auditing command authorization 56, 195
AUTH statement
 MSGRECV keyword usage 193
 usage 160
AUTHCHK keyword, OPTIONS
 statement 166
authorization checking
 commands 52, 196
 data set 63, 213
authorized receiver 161
AUTOCNT data set security 64, 214
automation operators
 determining active 149
automation router task 6, 7
 EMCS console name 25
 uniqueness of name 26, 34
automation task
 operator 161
autotask 6, 7
AUTOTBL data set security 64, 214
AUTOWRAP command 93

B

BACK command 93
BACKTBL keyword, OPTIONS
 statement 165
BEGIN statement 198
BLOG command 87, 94
BOTTOM command 94
BROWSE command 85, 94

C

CDS (couple data set) 9
CFRM couple data set 14
CFRM policy 14
 pending changes 123, 128
 policy switch 117
character translation sets 171
CHRON command 58, 209
CLOSE IMMED command,
 protecting 54
CMDAUTH keyword, OPTIONS
 statement 164
CNMSJ006 sample job 33
CNMSJM04 sample job 31
command
 containing special characters 56, 195
 restriction suggestions 55, 194
 security 52, 162, 196
command authorization
 auditing 56, 195
 EXCMD 62, 213
 MVS commands 62, 213
 source ID determination 57, 207
command authorization table
 auditing 56
command authorization types 51, 194
command entry area 80
command identifiers, NVSS command
 authorization table 196
commands
 ACF 147
 ALL 85, 93
 AOCTRACE 147
 AUTOWRAP 93
 BACK 93
 BLOG 87, 94
 BOTTOM 94
 BROWSE 85, 94
 DEFAULTS 95
 DISPACF 149
 DISPAOPS 149
 DISPASF 150
 DISPATHR 151
 DISPERRS 152
 DISPFK 95
 DISPFLGS 153
 DISPMGS 153
 DISPWTOR 154
 END 95
 FIND 85, 96

commands (*continued*)

 FORWARD 97
 HELP 97
 INDEX 97
 INGAUTO 155
 INGCF 106
 INGCF DRAIN 15, 111
 INGCF ENABLE 16, 116
 INGCF PATH 120
 INGCF STRUCTURE 121
 INGCFL 123
 INGCUST 155
 INGPLEX 124
 INGPLEX CDS 127
 INGPLEX CONSOLE 135
 INGPLEX IPL 137
 INGPLEX SDUMP 138
 INGPLEX SLIP 144
 INGPLEX SVCDUMP 141
 INGPLEX SYTEM 133
 INPUT 98
 issuing 80
 MVS commands 81
 LIST 98
 LISTA 98
 LOCATE 99
 LOGOFF 99
 MSG 99
 MVS 100
 not authority checked 55, 195
 OVERRIDE 100
 REPEAT 100
 repeating 81
 RETRIEVE 81, 101
 RETURN 101
 ROLL 83, 101
 SET 101
 TOP 102
 WHO 102
 WINDOW 102
COMMNDxx 29
COMMON section in AOFUCST 37
connector
 active 14
 failed persistent 15
CONSNAME keyword, PROFILE
 statement 170
console name, setting 170
couple data set (CDS) 9
 alternate CDS 9
 allocating 128
 recovery 9
 spare volumes 128
 turn into primary CDS 128
 policy 9
 primary CDS 9
coupling facility (CF) 14
 draining 15, 111
 enabling 16, 116, 117
 populating 117
 receiver path 107

coupling facility (CF) *(continued)*
 sender path 113, 117
 states 107

D

data set protection
 control viewing 66, 214
 general security 63, 213
 READSEC usage 66, 214
 data set security 63, 213
 debugging commands
 ACF 147
 AOCTRACE 147
 DISPACF 149
 DISPAOPS 149
 DISPASF 150
 DISPATHR 151
 DISPERRS 152
 DISPFLGS 153
 DISPMMSG 153
 DISPWTOR 154
 DEFAULTS command 95
 defining operator logon attributes 192
 definition
 statements
 msys for Operations 159
 definition statement format 159
 directory extent 11
 DISPACF command dialog 149
 DISPAOPS command 149
 DISPASF command 150
 DISPATHR command 151
 DISPERRS command 152
 DISPFK command 95
 DISPFLGS command 153
 DISPMMSG command 153
 DISPWTOR command 154
 domain name
 &DOMAIN symbol 33
 uniqueness 25, 33
 DSI802A message 69
 DSIDMN sample
 OPTIONS statement 163, 169
 TRANSTBL statement 171
 DSILIST DD statement 64, 214
 DSIOFP sample
 OPERATOR statement 161
 PROFILE statement 170
 DSIPARM member 31
 AOFCUST 31
 CxxSTYLE 31
 CNMSBAK1 31
 DSIDMNK 31
 DSIPRFAO sample
 PROFILE statement 169
 DSIPRFLC sample
 PROFILE statement 169
 DSIPRFMO sample
 PROFILE statement 169
 DSIPRFRD sample
 PROFILE statement 169
 DSIPROFA 169
 DSIPROFA sample
 AUTH statement 160
 PROFILE statement 169
 DSIPROFB 169

DSIPROFB sample
 PROFILE statement 169
 duplexing
 system-managed 122
 user-managed 122
 dynamic command authorization
 dynamic 196

E

EBCDIC character set 171
 EMCS console name
 automation router task 25
 END command 95
 END statement 160, 198
 ENQ section in AOFCUST 40
 EXCMD command authorization 62, 213
 EXEMPT statement 201

F

failed persistent connector 15
 FIND command 85, 96
 flags 155
 FORWARD command 97

G

generic characters 197
 generic security statements 203
 GETCONID
 console name 170
 GROUP statement 202

H

HCL keyword, PROFILE statement 169
 held-screen indicator, msys for
 Operations panel 77
 HELP command 97
 HW section in AOFCUST 44

I

IC keyword
 PROFILE statement 169
 identifier, operator 161
 immediate commands, protecting 54
 INDEX command 97
 INGALLC0 sample job 29
 INGAUTO command 155
 INGCF command 15, 106
 authorizations 106
 DRAIN 15, 111
 ENABLE 16, 116
 PATH 16, 120
 STRUCTURE 16, 121
 INGCFL command 123
 INGCUST command 155
 INGEMPF sample 28
 INGNVAP0 sample 31
 INGPLEX command 124
 CDS 10, 127
 CONSOLE 135
 IPL 137

INGPLEX command *(continued)*

SDUMP 138
 SLIP 144
 SVCDUMP 141
 SYSTEM 133
 INGPLOG0 sample 29
 INGRCCCHK, internal routine 53
 INGSCH0 sample 28
 INGVAM major node sample 32
 initialization style sheet 34
 naming scheme 34
 SSname keyword 34
 TOWER keyword 35
 INPUT command 98
 IODF 17, 107
 IRXANCHR table 28
 IXC102A section in AOFCUST 46

J

Japanese character translation 171
 jobs
 CNMSJ006 33
 CNMSJM04 31
 INGALLC0 29

K

kanji, character set 171
 katakana, character set 171
 keywords
 AUTHCHK, OPTIONS statement 166
 BACKTBL, OPTIONS statement 165
 CMDAUTH, OPTIONS
 statement 164
 CNMCON
 AUTH statement 160
 CONSNAME, PROFILE
 statement 170
 HCL, PROFILE statement 169
 IC
 PROFILE statement 169
 MOD
 TRANSTBL statement 171
 MSGRECVR, AUTH statement 160
 OPERSEC, OPTIONS statement 163
 PASSWORD
 OPERATOR statement 161
 SAFNODEC, OPTIONS
 statement 166

L

labels 159
 language processor environment msys for
 Operations 27
 LIST command 98
 LISTA command 98
 LOCATE command 99
 log stream 10
 log stream data set 11
 logmode table 32
 LOGOFF command 99
 logon
 security 162
 logon attributes 193

logon procedure 75
LOGPROF1 command list 26
LOGR couple data set 11

M

message
 authorized receiver 161
 unsolicited 160
message area 78
message automation 6
 MPFLSTxx 28
message automation table 6
message display screen 77
 command entry area 80
 message area 78
 response area 79
 session identification line 77
message origin, netlog 86
message types, netlog 86, 87
messages
 AOF532I 69
 AOF540I 69
 DSI802A 69
MINIMAL value of OPERSEC 192
MOD keyword
 TRANSTBL statement 171
MPFLSTxx 28
MSG command 99
MSGRECV keyword, AUTH
 statement 160, 193
MSOAPROC startup procedure 31, 33
msys for Operations
 coexistence with SA OS/390 1.3 22, 227
 commands, protecting 52
 defining operator attributes 193
 NetView System Services)
 part 4
 password security 192
 starting 69
 Sysplex Functions part 4
MVS command 100
 console name 170
MVS commands
 issuing from msys for Operations 81
MVS component
 MPF (message processing facility) 7
 XCF (Cross-system Coupling Facility) 5
 XES (Cross-system Extended Services) 5
MVS system commands
 protecting 62, 213

N

NETCMDS class 52, 165
netlog 83
 displaying 85
 entry structure 85
 log data sets 85
 message origin 86
 message types 86, 87
 printing inactive log 85

NetView segment of an SAF
 product 193
NetView System Services (NVSS) 4
NOMSG option 224
NVSS
 commands, protecting 196
NVSS (NetView System Services) 4
NVSS command authorization table
 auditing 195
 command identifiers 196
 definition 194
 example 206
 EXEMPT statement 201
 GROUP statement 202
 loading 205
 PERMIT statement 203
 PROTECT statement 200
 SETVAR statement 204
 special characters 56, 195
 statements 198
 syntax 198
 testing 205

O

online help 89
operator
 adding or deleting dynamically 194
 attributes 192
 authority, defining 160
 autotask 161
 changing security method 191
 identifier 161
 logon
 security 49, 163
 password security 49
 profile 170
 routing messages to 160
 testing definitions 194
operator attributes
 using msys for Operations 193
operator identifiers
 restrictions 49, 192
operator information
 defining operators 191
 logon attributes 191
 passwords 191
OPERATOR statement 161
OPERCMDS class of an SAF product 63, 213
OPERSEC keyword 192
OPERSEC keyword, OPTIONS
 statement 163
OPTIONS statement 162
 OPERSEC keyword 192
OST (operator station task) 6
outstanding WTORs
 displaying 154
OVERRIDE command 100

P

PAGE section in AOFCUST 38
panels
 AOCTRACE 147
 Authorized Message Receivers 154

panels (*continued*)
 Automation Operators 150
 Command Response 149, 150
 DISPAOPS 150
 DISPASF 150
 DISPATHR 151
 DISPERRS 152
 DISPFLGS 153
 DISPMGS 153
 DISPWTOR 154, 155
 INGAUTO 155
PARMLIB
 MPFLSTxx 28
 PROGxx 29
 SCHEDxx 28
partitioned data set 30
password authorization 49
PASSWORD keyword
 OPERATOR statement 161
password security 49
 operator 49
 using msys for Operations 192
pattern-matching characters 203
pause status indicator, msys for
 Operations panel 77
pending policy changes 123, 128
percent sign (%) generic character 203
PERMIT statement 203
persistent connection 14
persistent structure 14
policy 9
 CFRM 14
 switch 128
preference list 14
primary CDS 9
procedure
 HSAPIPLC 31
 INGPHOM 31
 INGPIPLC 31
 INGPIXCU 31
 MSAOAPROC 33
PROFILE statement
 CONSNAME keyword 170
 HCL keyword 170
 IC keyword 170, 193
 overview 169
 usage 193
PROFILE statement 170
PROGxx 29
program function keys 83
 displaying settings 83
PROTECT statement 200
protecting commands 52, 196

Q

QRYGLOBL data set security 64, 214

R

RACF security
 example of command security 53
 operator attributes 193
READSEC usage
 restricting access 66, 214
rebuild 15, 112, 122

- rebuild (*continued*)
 - system-managed 15, 112, 122
 - user-managed 15, 112, 122
- receiver path 107
- recovery
 - alternate CDS 9
 - customization 10
 - system log 12
 - customization 12
 - system logger
 - customization 11
 - directory shortage 11
 - VSAM share options 11
 - WTO(R) buffer shortage 12
 - customization 13
- recovery automation flag 155
- REFRESH command
 - OPERSEC keyword 192
- REPEAT command 100
- response area 79
- restricting
 - commands, keywords, and values 196
- restricting access
 - WRITESEC 66, 214
- restricting access to commands 51, 194
- RETRIEVE command 81, 101
- return codes 224
- RETURN command 101
- REXX
 - IRXANCHR table 28
 - language processor environment msys for Operations 27
- ROLL command 83, 101

S

- SAF command authorization
 - auditing 56, 195
 - example 53
 - NETCMDS resources 52
 - special characters 56
- SAFNODEC keyword, OPTIONS statement 166
- samples
 - AOFCUST 35
 - CNMSJ006 33
 - CNMSJM04 31
 - INGALLC0 29
 - INGEMPF 28
 - INGNVAP0 31
 - INGPROG0 29
 - INGSCHE0 28
 - INGVTAM 32
- SCHEDxx 28
- SECMIGR
 - data set security 214
- security
 - changing method for operators 191
 - characters in statements
 - generic 203
 - pattern-matching 203
 - wildcards 203
 - controlling access to commands 52, 196
 - data set 63, 213
 - EXCMD authorization 62, 213

- security (*continued*)
 - file 63, 213
 - immediate commands 54
 - member 63, 213
 - MVS commands 62, 213
 - NETCMDS class 52
 - NVSS command authorization table 196
 - operator
 - type of logon security 162
 - password 49
 - SAF command authorization 52
- sender path 113, 117
- session identification line 77
- SET command 101
- SETVAR statement 204
- SLIP TRAP 144
- SSIname keyword (CxxSTYLE) 34
- startup procedure 31, 33
- storage key 28
- structure 14
 - allocation 14
 - conditions 108, 123
 - deallocation 14, 112, 121
 - duplexing 15
 - system-managed 122
 - user-managed 122
 - pending policy changes 123, 128
 - persistent 14
 - preference list 14
 - rebuild 15, 112, 122
 - system-managed 15, 112, 122
 - user-managed 15, 112, 122
- SUBMIT command 63, 213
- symbol 22
 - &DOMAIN 33
 - &NV2I 34
 - &SQ1 34
 - &SQ2 34
 - &SYSCONE 25, 33
 - &VQ1 34
- syntax conventions 159
- Sysplex Functions 4
- system authorization facility product
 - ADDSD, RACF command 64, 214
 - APPL class 164
 - auditing command authorization 56, 195
 - command authorization overview 51
 - command security 52
 - NETCMDS class 52
 - NetView segment 193
 - OPERCMD class 63, 213
- system log 12
- system logger
 - directory extent 11
 - log stream 10
 - log stream data set 11
 - VSAM share options 11
 - LOGR couple data set 11
- system-managed duplexing 122
- system-managed rebuild 15, 112, 122

T

- target libraries 26
- task 6

- task (*continued*)
 - automation router task 6, 7, 25, 34
 - autotask 6, 7
 - OST (operator station task) 6
 - testing operator definitions 194
 - TOP command 102
 - TOWER keyword (CxxSTYLE) 35
- trace
 - with AOCTRACE command 147
- translation
 - character 171
- TRANSTBL statement 171

U

- unbalanced quote 221
- unsolicited messages, routing to an operator 160
- user-managed duplexing 122
- user-managed rebuild 15, 112, 122

V

- VIEW command processor
 - return code 224
- VSAM cluster 30
 - password 71
- VTAM 5
 - configuration list ATCCONxx 32
 - defining msys for Operations to 32
 - INGVTAM sample 32
 - logmode table 32
- VTAM major node
 - changing logmode table entries 32
 - changing the domain ID 32
 - changing the password 32
- VTAMLST data sets 30

W

- wait indicator, msys for Operations
 - panel 77
- WHO command 102
- wildcards in security statements 203
- WINDOW command 102
- WRITESEC usage
 - restricting access 66, 214
- WTO(R) buffer 12, 47
- WTORs
 - displaying outstanding WTORs 154

X

- XCF (Cross-system Coupling Facility) 5
- XES (Cross-system Extended Services) 5

Readers' Comments — We'd Like to Hear from You

z/OS
Managed System Infrastructure for Operations
Setting Up and Using
Version 1 Release 2

Publication No. SC33-7968-02

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Complete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to find	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to understand	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Well organized	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Applicable to your tasks	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name

Address

Company or Organization

Phone No.



Fold and Tape

Please do not staple

Fold and Tape



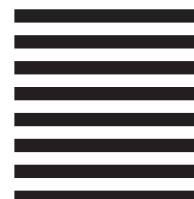
NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Deutschland Entwicklung GmbH
Information Development
Department 3248
Schoenaicher Strasse 220
71032 Boeblingen
Federal Republic of Germany



Fold and Tape

Please do not staple

Fold and Tape



Program Number: 5694-A01

Printed in U.S.A.

SC33-7968-02

